



SOAP API Developer Guide

Version 38.0, Winter '17



@salesforcedocs
Last updated: November 23, 2016

© Copyright 2000–2016 salesforce.com, inc. All rights reserved. Salesforce is a registered trademark of salesforce.com, inc., as are other names and marks. Other marks appearing herein may be trademarks of their respective owners.

CONTENTS

GETTING STARTED	1
Chapter 1: Introducing SOAP API	1
When to Use the SOAP API	2
Customize, Integrate, and Extend Your Salesforce Solutions	3
Supported Salesforce Editions	3
Standards Compliance	4
Development Platforms	4
SOAP API Support Policy	4
Choosing a WSDL	5
Related Resources	5
Quick Start	6
Step 1: Obtain a Salesforce Developer Edition Organization	6
Step 2: Generate or Obtain the Web Service WSDL	6
Step 3: Import the WSDL File Into Your Development Platform	7
Step 4: Walk Through the Sample Code	8
Chapter 2: Object Basics	30
Primitive Data Types	31
Field Types	32
Compound Fields	38
Address Compound Fields	38
Geolocation Compound Field	40
Compound Field Considerations and Limitations	42
API Data Types and Salesforce Field Types	43
Core Data Types Used in API Calls	44
sObject	44
API Fault Element	44
ExceptionCode	45
Error	49
StatusCode	50
ExtendedErrorDetails	60
Duplicate Management Data Types	60
System Fields	68
Required Fields	69
Frequently-Occurring Fields	70
API Field Properties	71
Relationships Among Objects	71
Relabeling Fields and Tabs and the API	72

Contents

Tooling API Objects in the Enterprise WSDL	73
Force.com AppExchange Object Prefixes and the API	73
Custom Objects	73
External Objects	76
Chapter 3: Call Basics	78
Characteristics of API Calls	79
Factors that Affect Data Access	79
Package Version Settings	81
Chapter 4: Error Handling	83
Error Handling for Session Expiration	84
More About Error Handling	84
Chapter 5: Security	85
User Authentication	86
User Profile and Permission Sets Configuration	86
Security Token	86
Sharing	87
Implicit Restrictions for Objects and Fields	88
API Access in Force.com AppExchange Packages	88
Outbound Port Restrictions	90
Chapter 6: Using the Partner WSDL	91
Obtaining the Partner WSDL File	92
Calls and the Partner WSDL	92
Objects, Fields, and Field Data and the Partner WSDL	93
Queries and the Partner WSDL	93
Namespaces in the Partner WSDL	94
Package Versions and the Partner WSDL	94
User Interface Themes	95
Examples Using the Partner WSDL	95
Sample query and queryMore Calls	99
Sample search Call	101
Sample create Call	104
Sample update Call	107
REFERENCE	110
Chapter 7: Data Model	110
Sales Objects	111
Task and Event Objects	112
Support Objects	113
Document, Note, and Attachment Objects	114
User, Sharing, and Permission Objects	115

Contents

Profile and Permission Objects	116
Record Type Objects	117
Product and Schedule Objects	118
Sharing and Team Selling Objects	119
Customizable Forecasting Objects	119
Forecasts Objects	120
Territory Management 2.0 Objects	120
Territory Management	122
Process Objects	123
Content Objects	124
ContentNote Objects	124
Chatter Objects	125
Chatter Feed Objects	127
Salesforce Knowledge Objects	128
Work.com Badge and Reward Objects	129
Work.com Feedback and Performance Cycle Objects	130
Chapter 8: Standard Objects	131
AcceptedEventRelation	168
Account	169
AccountContactRelation	189
AccountCleanInfo	191
AccountContactRole	213
AccountFeed	214
AccountHistory	222
AccountOwnerSharingRule	223
AccountPartner	226
AccountShare	228
AccountTag	232
AccountTeamMember	233
AccountTerritoryAssignmentRule	237
AccountTerritoryAssignmentRuleItem	238
AccountTerritorySharingRule	240
ActionLinkGroupTemplate	243
ActionLinkTemplate	245
ActivityHistory	250
AdditionalNumber	258
AgentWork	259
AllowedEmailDomain	265
Announcement	266
ApexClass	268
ApexComponent	270
ApexLog	273
ApexPage	275

Contents

ApexTestQueueItem	278
ApexTestResult	280
ApexTestResultLimits	283
ApexTestRunResult	286
ApexTestSuite	289
ApexTrigger	289
AppMenuItem	293
Approval	299
Article Type__DataCategorySelection	301
Article Type__Feed	303
Asset	309
AssetFeed	313
AssetOwnerSharingRule	321
AssetShare	323
AssetTag	324
AssignedResource	326
AssignmentRule	327
AsyncApexJob	328
AttachedContentDocument	331
AttachedContentNote	333
Attachment	335
AuraDefinition	339
AuraDefinitionBundle	341
AuthConfig	343
AuthConfigProviders	346
AuthProvider	347
AuthSession	352
BackgroundOperation	355
BackgroundOperationResult	360
Bookmark	362
BrandTemplate	363
BusinessHours	365
BusinessProcess	369
CallCenter	371
Campaign	373
CampaignFeed	381
CampaignInfluence	389
CampaignInfluenceModel	390
CampaignMember	392
CampaignMemberStatus	399
CampaignOwnerSharingRule	401
CampaignShare	403
CampaignTag	404
Case	406

Contents

CaseArticle	416
CaseComment	417
CaseContactRole	420
CaseFeed	421
CaseHistory	429
CaseMilestone	430
CaseOwnerSharingRule	434
CaseShare	436
CaseSolution	438
CaseStatus	439
CaseTag	440
CaseTeamMember	442
CaseTeamRole	443
CaseTeamTemplate	443
CaseTeamTemplateMember	444
CaseTeamTemplateRecord	445
CategoryData	446
CategoryNode	447
CategoryNodeLocalization	448
ChatterActivity	453
ChatterAnswersActivity	455
ChatterAnswersReputationLevel	459
ChatterConversation	460
ChatterConversationMember	461
ChatterMessage	462
ClientBrowser	463
CollaborationGroup	465
CollaborationGroupFeed	471
CollaborationGroupMember	478
CollaborationGroupMemberRequest	480
CollaborationGroupRecord	482
CollaborationInvitation	483
CombinedAttachment	486
Community (Zone)	488
ConnectedApplication	490
Contact	492
ContactCleanInfo	504
ContactFeed	514
ContactHistory	521
ContactOwnerSharingRule	522
ContactShare	524
ContactTag	526
ContentAsset	528
ContentDistribution	529

Contents

ContentDistributionView	535
ContentDocument	536
ContentDocumentFeed	542
ContentDocumentHistory	549
ContentDocumentLink	551
ContentFolder	555
ContentFolderItem	555
ContentFolderLink	557
ContentFolderMember	558
ContentHubItem	559
ContentHubRepository	563
ContentNote	564
ContentVersion	568
ContentVersionHistory	579
ContentWorkspace	581
ContentWorkspaceDoc	582
Contract	584
ContractContactRole	593
ContractFeed	594
ContractHistory	602
ContractLineItem	603
ContractLineItemHistory	607
ContractStatus	609
ContractTag	611
CronJobDetail	612
CronTrigger	613
CurrencyType	616
CustomBrand	617
CustomBrandAsset	618
Custom Metadata Type__mdt	621
Custom Object__Feed	623
CustomPermission	630
CustomPermissionDependency	633
DandBCompany	635
Dashboard	658
DashboardComponent	663
DashboardComponentFeed	664
DashboardFeed	672
DashboardTag	679
DataAssessmentFieldMetric	680
DataAssessmentMetric	682
DataAssessmentValueMetric	683
DatacloudCompany	684
DatacloudContact	693

Contents

DatacloudDandBCompany	698
DatacloudOwnedEntity	721
DatacloudPurchaseUsage	723
DatacloudSocialHandle	725
DatedConversionRate	726
DcSocialProfile	727
DcSocialProfileHandle	729
DeclinedEventRelation	730
Division	732
DivisionLocalization	733
Document	735
DocumentAttachmentMap	739
DocumentTag	740
Domain	742
DomainSite	743
DuplicateRecordItem	744
DuplicateRecordSet	745
DuplicateRule	747
EmailDomainKey	749
EmailMessage	751
EmailMessageRelation	757
EmailServicesAddress	759
EmailServicesFunction	761
EmailStatus	766
EmailTemplate	767
Entitlement	772
EntitlementContact	776
EntitlementFeed	777
EntitlementHistory	785
EntitlementTemplate	786
EntityHistory	788
EntityMilestone	790
EntitySubscription	797
EnvironmentHubMember	799
Event	803
EventFeed	817
EventLogFile	824
EventRelation	948
EventTag	953
EventWhoRelation	954
ExternalDataSource	955
ExternalDataUserAuth	958
FeedAttachment	960
FeedComment	963

Contents

FeedItem	968
FeedLike	978
FeedPollChoice	979
FeedPollVote	981
FeedPost	981
FeedRevision	985
FeedTrackedChange	988
FieldHistoryArchive	991
FieldPermissions	994
FiscalYearSettings	997
FlexQueueItem	1000
FlowInterview	1002
FlowInterviewOwnerSharingRule	1003
FlowInterviewShare	1005
Folder	1006
FolderedContentDocument	1009
ForecastingAdjustment	1010
ForecastingFact	1015
ForecastingItem	1017
ForecastingOwnerAdjustment	1023
ForecastingQuota	1027
ForecastingType	1030
ForecastShare	1032
Goal	1033
GoalFeed	1036
GoalHistory	1041
GoalLink	1042
GoalShare	1043
GoogleDoc	1045
Group	1046
GroupMember	1049
HashtagDefinition	1050
Holiday	1052
Idea	1055
IdeaComment	1061
IdeaReputation	1063
IdeaReputationLevel	1066
IdeaTheme	1067
KnowledgeableUser	1069
KnowledgeArticle	1070
KnowledgeArticleVersion	1072
KnowledgeArticleVersionHistory	1079
KnowledgeArticleViewStat	1082
KnowledgeArticleVoteStat	1084

Contents

Lead	1085
LeadCleanInfo	1103
LeadFeed	1117
LeadHistory	1124
LeadOwnerSharingRule	1126
LeadShare	1128
LeadStatus	1130
LeadTag	1131
LimitAllocationPerApp	1133
LineitemOverride	1135
LinkedArticle	1137
LinkedArticleFeed	1139
LinkedArticleHistory	1142
ListView	1143
ListViewChart	1146
ListViewChartInstance	1147
LiveAgentSession	1151
LiveAgentSessionHistory	1154
LiveAgentSessionOwnerSharingRule	1156
LiveAgentSessionShare	1157
LiveChatBlockingRule	1159
LiveChatButton	1161
LiveChatButtonDeployment	1168
LiveChatButtonSkill	1169
LiveChatDeployment	1170
LiveChatSensitiveDataRule	1173
LiveChatTranscript	1176
LiveChatTranscriptEvent	1183
LiveChatTranscriptHistory	1185
LiveChatTranscriptOwnerSharingRule	1186
LiveChatTranscriptShare	1188
LiveChatTranscriptSkill	1189
LiveChatUserConfig	1190
LiveChatUserConfigProfile	1197
LiveChatUserConfigUser	1198
LiveChatVisitor	1198
LoginEvent	1199
LoginGeo	1207
LoginHistory	1209
LoginIp	1214
LookedUpFromActivity	1215
MacroInstruction	1222
MailmergeTemplate	1224
MatchingRule	1226

Contents

MatchingRuleItem	1229
MetadataPackage	1230
MetadataPackageVersion	1231
Metric	1234
MetricDataLink	1238
MetricDataLinkHistory	1239
MetricFeed	1240
MetricHistory	1245
MetricsDataFile	1246
MetricShare	1249
MilestoneType	1251
Name	1252
NamedCredential	1255
NavigationLinkSet	1258
NavigationMenuItem	1259
NavigationMenuItemLocalization	1261
Network	1263
NetworkActivityAudit	1271
NetworkMember	1273
NetworkMemberGroup	1279
NetworkModeration	1280
NetworkPageOverride	1282
NetworkSelfRegistration	1283
NewsFeed	1284
Note	1291
OauthToken	1292
NoteAndAttachment	1295
NoteTag	1296
ObjectPermissions	1298
ObjectTerritory2AssignmentRule	1301
ObjectTerritory2AssignmentRuleItem	1303
ObjectTerritory2Association	1304
OpenActivity	1306
OperatingHours	1313
OperatingHoursFeed	1315
OperatingHoursHistory	1317
Opportunity	1318
OpportunityCompetitor	1329
OpportunityContactRole	1330
OpportunityFeed	1332
OpportunityFieldHistory	1339
OpportunityHistory	1341
OpportunityLineItem	1343
OpportunityLineItemSchedule	1349

Contents

OpportunityOverride	1353
OpportunityOwnerSharingRule	1356
OpportunityPartner	1358
OpportunityShare	1360
OpportunitySplit	1362
OpportunitySplitType	1364
OpportunityStage	1367
OpportunityTag	1369
OpportunityTeamMember	1370
Order	1373
OrderFeed	1383
OrderHistory	1389
OrderItem	1391
OrderItemFeed	1393
OrderItemHistory	1400
OrderOwnerSharingRule	1401
Organization	1404
OrgWideEmailAddress	1424
OwnedContentDocument	1425
PackageLicense	1427
PackagePushError	1429
PackagePushJob	1431
PackagePushRequest	1434
PackageSubscriber	1436
Partner	1440
PartnerNetworkConnection	1443
PartnerNetworkRecordConnection	1446
PartnerNetworkSyncLog	1449
PartnerRole	1451
Period	1453
PermissionSet	1455
PermissionSetAssignment	1461
PermissionSetLicense	1462
PermissionSetLicenseAssign	1465
PlatformAction	1466
PresenceUserConfig	1473
PresenceUserConfigProfile	1476
PresenceUserConfigUser	1477
Pricebook2	1477
Pricebook2History	1481
PricebookEntry	1482
ProcessDefinition	1485
ProcessInstance	1487
ProcessInstanceHistory	1489

Contents

ProcessInstanceStep	1492
ProcessInstanceNode	1495
ProcessInstanceWorkitem	1497
ProcessNode	1499
Product2	1500
Product2Feed	1506
ProductEntitlementTemplate	1514
Profile	1515
ProfileSkill	1518
ProfileSkillEndorsement	1520
ProfileSkillEndorsementFeed	1521
ProfileSkillEndorsementHistory	1525
ProfileSkillFeed	1526
ProfileSkillHistory	1532
ProfileSkillShare	1533
ProfileSkillUser	1534
ProfileSkillUserFeed	1535
ProfileSkillUserHistory	1540
PushTopic	1541
QuantityForecast	1543
QuantityForecastHistory	1551
QueueRoutingConfig	1553
Question	1556
QuestionDataCategorySelection	1560
QuestionReportAbuse	1562
QuestionSubscription	1563
QueueSobject	1565
QuickText	1566
QuickTextHistory	1568
QuickTextOwnerSharingRule	1569
QuickTextShare	1571
Quote	1572
QuoteDocument	1585
QuoteLineItem	1587
RecentlyViewed	1591
RecordType	1595
RecordTypeLocalization	1598
Reply	1599
ReplyReportAbuse	1602
Report	1603
ReportFeed	1607
ReportTag	1614
ReputationLevel	1616
ReputationLevelLocalization	1617

Contents

ReputationPointsRule	1619
ResourceAbsence	1620
ResourceAbsenceFeed	1624
ResourceAbsenceHistory	1626
RevenueForecast	1627
RevenueForecastHistory	1636
RuleTerritory2Association	1638
SamlSsoConfig	1639
Scontrol	1644
ScontrolLocalization	1647
SearchPromotionRule	1653
SecureAgent	1654
SecureAgentsCluster	1655
SelfServiceUser	1657
ServiceAppointment	1660
ServiceAppointmentFeed	1668
ServiceAppointmentHistory	1670
ServiceAppointmentOwnerSharingRule	1671
ServiceAppointmentShare	1673
ServiceChannel	1674
ServiceChannelStatus	1676
ServiceContract	1677
ServiceContractFeed	1685
ServiceContractHistory	1693
ServiceContractOwnerSharingRule	1694
ServiceContractShare	1696
ServicePresenceStatus	1698
ServiceResource	1699
ServiceResourceCapacity	1702
ServiceResourceCapacityFeed	1705
ServiceResourceCapacityHistory	1707
ServiceResourceFeed	1708
ServiceResourceHistory	1711
ServiceResourceOwnerSharingRule	1712
ServiceResourceShare	1714
ServiceResourceSkill	1715
ServiceResourceSkillFeed	1717
ServiceResourceSkillHistory	1719
ServiceTerritory	1720
ServiceTerritoryFeed	1725
ServiceTerritoryHistory	1727
ServiceTerritoryMember	1728
ServiceTerritoryMemberFeed	1733
ServiceTerritoryMemberHistory	1735

Contents

SessionPermSetActivation	1736
SetupAuditTrail	1738
SetupEntityAccess	1740
SignupRequest	1742
Site	1749
SiteDomain	1755
SiteHistory	1756
Skill	1757
SkillProfile	1759
SkillRequirement	1759
SkillRequirementFeed	1761
SkillRequirementHistory	1764
SkillUser	1765
SlaProcess	1766
Solution	1769
SolutionFeed	1773
SolutionHistory	1780
SolutionStatus	1782
SolutionTag	1783
SOSDeployment	1785
SOSSession	1787
SOSSessionActivity	1790
SOSSessionHistory	1791
SOSSessionOwnerSharingRule	1792
SOSSessionShare	1794
StaticResource	1796
StreamingChannel	1798
TagDefinition	1800
Task	1801
TaskFeed	1812
TaskPriority	1820
TaskRelation	1821
TaskStatus	1823
TaskTag	1825
TaskWhoRelation	1826
TenantSecret	1828
Territory	1830
Territory2	1833
Territory2Model	1835
Territory2ModelHistory	1837
Territory2Type	1838
TestSuiteMembership	1840
ThirdPartyAccountLink	1841
TimeSlot	1843

Contents

TimeSlotHistory	1845
Topic	1846
TopicAssignment	1848
TopicFeed	1850
TopicLocalization—Beta	1856
TwoFactorInfo	1861
TwoFactorMethodInfo	1862
TwoFactorTempCode	1864
UndecidedEventRelation	1865
User	1866
UserAccountTeamMember	1907
UserApplInfo	1910
UserAppMenuCustomization	1911
UserAppMenuCustomizationShare	1912
UserAppMenuItem	1913
UserConfigTransferButton	1917
UserConfigTransferSkill	1917
UserCustomBadge	1918
UserCustomBadgeLocalization	1919
UserFeed	1921
UserLicense	1929
UserLogin	1932
UserMembershipSharingRule	1933
UserPackageLicense	1935
UserPreference	1936
UserProfile	1937
UserProfileFeed	1955
UserProvAccount	1962
UserProvAccountStaging	1965
UserProvMockTarget	1968
UserProvisioningConfig	1970
UserProvisioningLog	1974
UserProvisioningRequest	1975
UserProvisioningRequestOwnerSharingRule	1981
UserProvisioningRequestShare	1982
UserRecordAccess	1984
UserRole	1986
UserServicePresence	1990
UserShare	1991
UserTeamMember	1993
UserTerritory	1995
UserTerritory2Association	1996
VerificationHistory	1997
Vote	2003

Contents

WebLink	2004
WebLinkLocalization	2009
WorkAccess	2014
WorkAccessShare	2015
WorkBadge	2016
WorkBadgeDefinition	2019
WorkBadgeDefinitionHistory	2022
WorkBadgeDefinitionShare	2023
WorkCoaching	2025
WorkCoachingFeed	2026
WorkCoachingHistory	2032
WorkCoachingShare	2033
WorkFeedback	2035
WorkFeedbackHistory	2036
WorkFeedbackQuestion	2037
WorkFeedbackQuestionHistory	2040
WorkFeedbackQuestionSet	2041
WorkFeedbackQuestionSetHistory	2042
WorkFeedbackQuestionSetShare	2043
WorkFeedbackQuestionShare	2044
WorkFeedbackRequest	2046
WorkFeedbackRequestFeed	2051
WorkFeedbackRequestHistory	2056
WorkFeedbackRequestShare	2057
WorkFeedbackShare	2059
WorkGoal	2060
WorkGoalCollaborator	2066
WorkGoalCollaboratorHistory	2067
WorkGoalFeed	2069
WorkGoalHistory	2075
WorkGoalLink	2076
WorkGoalShare	2077
WorkOrder	2078
WorkOrderFeed	2088
WorkOrderHistory	2092
WorkOrderLineItem	2093
WorkOrderLineItemFeed	2101
WorkOrderLineItemHistory	2105
WorkOrderShare	2106
WorkPerformanceCycle	2107
WorkPerformanceCycleFeed	2110
WorkPerformanceCycleHistory	2115
WorkPerformanceCycleShare	2116
WorkReward	2118

Contents

WorkRewardFund	2120
WorkRewardFundHistory	2122
WorkRewardFundShare	2123
WorkRewardFundType	2124
WorkRewardFundTypeHistory	2127
WorkRewardFundTypeShare	2128
WorkRewardHistory	2130
WorkRewardShare	2131
WorkThanks	2132
WorkThanksShare	2133
WorkType	2134
WorkTypeFeed	2136
WorkTypeHistory	2139
Reusable Terms for API Objects	2140
Chapter 9: Core Calls	2141
convertLead()	2142
LeadConvertResult	2148
create()	2149
SaveResult	2158
delete()	2159
DeleteResult	2162
emptyRecycleBin()	2162
EmptyRecycleBinResult	2165
executeListView()	2165
ExecuteListViewRequest	2166
ExecuteListViewResult	2166
ListViewColumn	2167
ListViewRecord	2167
ListViewRecordColumn	2168
findDuplicates()	2168
getDeleted()	2172
GetDeletedResult	2176
getUpdated()	2176
GetUpdatedResult	2180
invalidateSessions()	2180
InvalidateSessionsResult	2182
login()	2182
LoginResult	2187
logout()	2188
merge()	2189
MergeResult	2194
performQuickActions()	2195
PerformQuickActionResult	2196

Contents

process()	2196
ProcessResult	2199
query()	2200
QueryResult	2203
QueryLocator	2206
queryAll()	2206
queryMore()	2209
QueryResult	2212
QueryLocator	2213
retrieve()	2213
search()	2216
SearchResult	2219
undelete()	2219
UndeleteResult	2222
update()	2223
SaveResult	2229
upsert()	2230
UpsertResult	2235
Chapter 10: Describe Calls	2236
describeAllTabs()	2237
describeAppMenu()	2238
DescribeAppMenuResult	2239
describeApprovalLayout()	2240
DescribeApprovalLayoutResult	2242
describeAvailableQuickActions()	2242
DescribeAvailableQuickActionResult	2243
describeCompactLayouts()	2244
DescribeCompactLayoutsResult	2246
describeDataCategoryGroups()	2247
DescribeDataCategoryGroupResult	2249
describeDataCategoryGroupStructures()	2250
describeDataCategoryGroupStructures()	2254
describeFlexiPages()	2254
DescribeFlexiPageResult	2256
describeGlobal()	2259
DescribeGlobalResult	2261
describeGlobalTheme()	2263
DescribeGlobalTheme	2264
describeKnowledge()Call Name	2265
describeLayout()	2266
DescribeLayoutResult	2271
describePrimaryCompactLayouts()	2283
describeQuickActions()	2285

Contents

DescribeQuickActionResult	2286
describeSearchScopeOrder()	2291
DescribeSearchScopeOrderResult	2292
describeSearchLayouts()	2292
DescribeSearchLayoutResult	2294
describeSObject()	2294
describeSObjectResult	2297
describeSObjects()	2298
DescribeSObjectResult	2302
describeSoftphoneLayout()	2314
describeSoqlListViews()	2318
DescribeSoqlListView	2319
DescribeSoqlListViewParams	2319
DescribeSoqlListViewResult	2319
DescribeSoqlListViewsRequest	2320
ListViewColumn	2320
ListViewOrderBy	2321
describeTabs()	2321
describeTabSetResult	2324
describeTheme()	2326
DescribeThemeResult	2327
DescribeThemeItem	2328
Chapter 11: Utility Calls	2329
getServerTimestamp()	2329
getServerTimestampResult	2331
getUserInfo()	2331
getUserInfoResult	2333
renderEmailTemplate()	2334
RenderEmailTemplateResult	2337
resetPassword()	2338
sendEmail()	2340
SendEmailResult	2349
sendEmailMessage()	2350
setPassword()	2353
Chapter 12: SOAP Headers	2355
AllOrNoneHeader	2356
AllowFieldTruncationHeader	2357
AssignmentRuleHeader	2359
CallOptions	2360
DisableFeedTrackingHeader	2361
DuplicateRuleHeader	2362
EmailHeader	2363

Contents

LimitInfoHeader	2365
LocaleOptions	2366
LoginScopeHeader	2367
MruHeader	2368
OwnerChangeOptions	2369
PackageVersionHeader	2372
QueryOptions	2373
SessionHeader	2374
UserTerritoryDeleteHeader	2374
USING THE API WITH SALESFORCE FEATURES	2375
Chapter 13: Implementation Considerations	2375
Choosing a User for an Integration	2376
Login Server URL	2376
Log in to the Login Server	2377
Typical API Call Sequence	2377
Salesforce Sandbox	2377
Multiple Instances of Salesforce Database Servers	2378
Content Type Requirement	2378
Monitoring API Traffic	2378
API Usage Metering	2378
Compression	2380
HTTP Persistent Connections	2381
HTTP Chunking	2381
Internationalization and Character Sets	2381
XML Compliance	2381
.NET, Non-String Fields, and the Enterprise WSDL	2382
Chapter 14: Outbound Messaging	2383
Understanding Outbound Messaging	2384
Understanding Notifications	2385
Setting Up Outbound Messaging	2385
Setting Up User Profiles	2386
Defining Outbound Messaging	2386
Downloading the Salesforce Client Certificate	2387
Viewing Outbound Messages	2387
Tracking Outbound Message Status	2388
Considerations for Security	2388
Understanding the Outbound Messaging WSDL	2388
Building a Listener	2390
Chapter 15: Data Loading and Integration	2392
Choosing the Right API	2393

Contents

Client Application Design	2393
Salesforce Settings	2394
Best Practices with Any Data Loader	2394
Integration and Single Sign-On	2395
Chapter 16: Data Replication	2396
API Calls for Data Replication	2397
Scope of Data Replication	2397
Data Replication Steps	2397
Object-Specific Requirements for Data Replication	2398
Polling for Changes	2398
Checking for Structural Changes in the Object	2399
Chapter 17: Feature-Specific Considerations	2400
Archived Activities	2401
Person Account Record Types	2401
Opportunity Forecast Override Business Rules	2402
OpportunityOverride Lifecycle	2403
LineitemOverride Object Lifecycle	2405
External Objects	2406
Call Centers and the API	2406
Implementing Salesforce Integrations on Force.com	2408
Articles	2409
Data Categories	2412
GLOSSARY	2415
INDEX	2431

GETTING STARTED

CHAPTER 1 Introducing SOAP API

In this chapter ...

- When to Use the SOAP API
- Customize, Integrate, and Extend Your Salesforce Solutions
- Supported Salesforce Editions
- Standards Compliance
- Development Platforms
- SOAP API Support Policy
- Choosing a WSDL
- Related Resources
- Quick Start

Salesforce provides programmatic access to your org's information using simple, powerful, and secure application programming interfaces. To use this document, you should have a basic familiarity with software development, Web services, and the Salesforce user interface.

Any functionality described in this guide is available if your organization has the API feature enabled. This feature is enabled by default for Performance, Unlimited, Enterprise, and Developer Editions. Some Professional Edition organizations may also have the API enabled. If you cannot access the features you see in this guide, contact Salesforce.



Note: Salesforce Education Services offers a suite of training courses to enable developers to design, create, integrate, and extend applications built on the Force.com platform. Be sure to visit <http://www.salesforce.com/training> to learn more.

When to Use the SOAP API

The Salesforce prebuilt applications provide powerful CRM functionality. In addition, Salesforce provides the ability to customize the prebuilt applications to fit your organization. However, your organization may have complex business processes that are unsupported by the existing functionality. When this is the case, the Force.com platform includes a number of ways for advanced administrators and developers to implement custom functionality. These include the SOAP API, Apex, and Visualforce.

SOAP API

Use SOAP API to create, retrieve, update or delete records, such as accounts, leads, and custom objects. With more than 20 different calls, SOAP API also allows you to maintain passwords, perform searches, and much more. Use SOAP API in any language that supports Web services.

REST API

REST API provides a powerful, convenient, and simple REST-based web services interface for interacting with Salesforce. Its advantages include ease of integration and development, and it's an excellent choice of technology for use with mobile applications and web projects. However, if you have many records to process, consider using Bulk API, which is based on REST principles and optimized for large sets of data.

Bulk API

Bulk API is based on REST principles and is optimized for loading or deleting large sets of data. You can use it to query, insert, update, upsert, or delete many records asynchronously by submitting batches. Salesforce processes batches in the background.

SOAP API, in contrast, is optimized for real-time client applications that update a few records at a time. You can use SOAP API for processing many records, but when the data sets contain hundreds of thousands of records, SOAP API is less practical. Bulk API is designed to make it simple to process data from a few thousand to millions of records.

Metadata API

Use Metadata API to retrieve, deploy, create, update, or delete customizations for your org. The most common use is to migrate changes from a sandbox or testing org to your production environment. Metadata API is intended for managing customizations and for building tools that can manage the metadata model, not the data itself.

The easiest way to access the functionality in Metadata API is to use the Force.com IDE or Force.com Migration Tool. Both tools are built on top of Metadata API and use the standard Eclipse and Ant tools, respectively, to simplify working with Metadata API.

- Force.com IDE is built on the Eclipse platform, for programmers familiar with integrated development environments. Code, compile, test, and deploy from within the IDE.
- The Force.com Migration Tool is ideal if you use a script or the command line for moving metadata between a local directory and a Salesforce org.

Apex

Use Apex if you want to:

- Create Web services.
- Create email services.

- Perform complex validation over multiple objects.
- Create complex business processes that are not supported by workflow.
- Create custom transactional logic (logic that occurs over the entire transaction, not just with a single record or object).
- Attach custom logic to another operation, such as saving a record, so that it occurs whenever the operation is executed, regardless of whether it originates in the user interface, a Visualforce page, or from SOAP API.

For more information, see the [Apex Developer Guide](#).

Visualforce

Visualforce consists of a tag-based markup language that gives developers a more powerful way of building applications and customizing the Salesforce user interface. With Visualforce you can:

- Build wizards and other multistep processes.
- Create your own custom flow control through an application.
- Define navigation patterns and data-specific rules for optimal, efficient application interaction.

For more information, see the [Visualforce Developer's Guide](#).

Customize, Integrate, and Extend Your Salesforce Solutions

The Force.com platform allows you to customize, integrate, and extend your Salesforce organization using the language and platform of your choice:

- **Customize Salesforce** with custom fields, links, objects, page layouts, buttons, record types, s-controls, and tabs to meet specific business requirements.
- **Integrate Salesforce** with your organization's ERP and finance systems, deliver real-time sales and support information to company portals, and populate critical business systems with customer information.
- **Extend Salesforce** in presentation, business logic, and data services with new functionality that reflects the business requirements of your organization.

For more information about Force.com solutions, developer resources, and community resources, go to [Salesforce Developers](#).

Supported Salesforce Editions

To use SOAP API, your organization must use Enterprise Edition, Performance Edition, Unlimited Edition, or Developer Edition. If you are an existing Salesforce customer and want to upgrade to Enterprise, Unlimited, or Performance Edition, contact your account representative.

To develop Web service client applications, it is strongly recommended that you use Developer Sandbox, which is an exact replica of your Salesforce deployment, including all customization and data. For more information, see

<http://www.salesforce.com/products/sandbox.jsp>.

Developer Edition provides access to all of the features available with Enterprise Edition. Developer Edition is constrained only by the number of users and the amount of storage space. Developer Edition provides a development context that allows you to build and test your solutions without affecting your organization's live data. Developer Edition accounts are available for free at

https://developer.salesforce.com/page/Getting_Started.

Standards Compliance

SOAP API is implemented to comply with the following specifications:

Standard Name	Website
Simple Object Access Protocol (SOAP) 1.1	
Web Service Description Language (WSDL) 1.1	http://www.w3.org/TR/2001/NOTE-wsdl-20010315
WS-I Basic Profile 1.1	http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html

Development Platforms

SOAP API works with current SOAP development environments, including, but not limited to, Visual Studio .NET 2005. In this document, we provide examples in Java and C# (.NET). The Java examples are based on WSC 20.0 (WSC) and JDK 6 (Java Platform Standard Edition Development Kit 6). Additional versions of WSC are available at <https://github.com/forcedotcom/wsc> and <http://mvnrepository.com/artifact/com.force.api/force-wsc>. To see a complete list of compatible development platforms and more sample code, go to developer.salesforce.com.



Note: Development platforms vary in their SOAP implementations. Implementation differences in certain development platforms might prevent access to some or all of the features in the API. If you are using Visual Studio for .NET development, we recommend that you use Visual Studio 2003 or higher.

SOAP API Support Policy

Salesforce recommends that your new client applications use the most recent version of the Force.com WSDL file to fully exploit the benefits of richer features and greater efficiency. You can navigate to the most recent WSDL for your organization from Setup by entering "API" in the Quick Find box, then selecting **API**. When a new version is released, use the following steps in [Quick Start](#) to update your WSDL:

- Regenerate the WSDL file (see [Step 2: Generate or Obtain the Web Service WSDL](#))
- Import it into your environment (see [Step 3: Import the WSDL File Into Your Development Platform](#))

Backward Compatibility

Salesforce strives to make backward compatibility easy when using the Force.com platform.

Each new Salesforce release consists of two components:

- A new release of platform software that resides on Salesforce systems
- A new version of SOAP API

For example, the Winter '07 release included SOAP API version 9.0 and the Summer '07 release included SOAP API version 10.0.

We maintain support for each SOAP API version across releases of the platform software. SOAP API is backward compatible in that an application created to work with a given SOAP API version will continue to work with that same SOAP API version in future platform software releases.

Salesforce does not guarantee that an application written against one SOAP API version will work with future SOAP API versions: Changes in method signatures and data representations are often required as we continue to enhance SOAP API. However, we strive to keep SOAP API consistent from version to version with minimal if any changes required to port applications to newer SOAP API versions.

For example, an application written using SOAP API version 9.0 which shipped with the Winter '07 release will continue to work with SOAP API version 9.0 on the Summer '07 release and on future releases beyond that. However, that same application may not work with SOAP API version 10 without modifications to the application.

SOAP API End-of-Life

Salesforce is committed to supporting each SOAP API version for a minimum of three years from the date of first release. In order to improve the quality and performance of SOAP API, versions that are more than three years old may cease to be supported.

When a SOAP API version is scheduled to be unsupported, an advance end-of-life notice will be given at least one year before support for SOAP API version is ended. Salesforce will directly notify customers using SOAP API versions scheduled for end of life.

Choosing a WSDL

There are two Force.com Web services for which you can obtain WSDL files for API access:

- **Force.com Enterprise WSDL**—This API is for most enterprise users who are developing client applications for their organization. The enterprise WSDL file is a strongly typed representation of your organization's data. It provides information about your schema, data types, and fields to your development environment, allowing for a tighter integration between it and the Force.com Web service. This WSDL changes if custom fields or custom objects are added to, renamed, or removed from, your organization's Salesforce configuration. If you are downloading an enterprise WSDL and you have managed packages installed in your organization, you need to take an extra step to select the version of each installed package to include in the generated WSDL.
Note the following when generating the enterprise WSDL:
 - If new custom fields or objects are added to, renamed, or removed from your organization's information, you need to regenerate the WSDL file in order to access them.
 - The generated WSDL contains the objects and fields in your organization, including those available in the selected versions of each installed package. If a field or object is added in a later package version, you must generate the enterprise WSDL with that package version to work with the object or field in your API integration.
- **Force.com Partner WSDL**—This API is for Salesforce partners who are developing client applications for multiple organizations. As a loosely-typed representation of the Salesforce object model, the [partner WSDL](#) can be used to access data within any organization.

Related Resources

The Salesforce developer website provides a full suite of developer toolkits, sample code, sample SOAP messages, community-based support, and other resources to help you with your development projects. Be sure to visit

https://developer.salesforce.com/page/Getting_Started for more information, or visit

<https://developer.salesforce.com/signup> to sign up for a free Developer Edition account.

You can visit these websites to find out more about Salesforce applications:

- [Salesforce](#) for information about the Salesforce application.
- [Force.com AppExchange](#) for access to apps created for Salesforce.
- [Salesforce.com Community](#) for services to ensure Salesforce customer success.

Quick Start

Use this quick start to create a sample application in your development environment.



Note: Before you begin building an integration or other client application:

- Install your development platform according to its product documentation.
- Read through all the steps before beginning this quick start. You may also wish to review the rest of this document to familiarize yourself with terms and concepts.

Step 1: Obtain a Salesforce Developer Edition Organization

If you are not already a member of the Force.com developer community, go to developer.salesforce.com/signup and follow the instructions for signing up for a Developer Edition organization. Even if you already have Enterprise Edition, Unlimited Edition, or Performance Edition, use Developer Edition for developing, staging, and testing your solutions against sample data to protect your organization's live data. This is especially true for applications that insert, update, or delete data (as opposed to simply reading data).

If you already have a Developer Edition organization, verify that you have the "API Enabled" permission. This permission is enabled by default, but may have been changed by an administrator. For more information, see the help in the Salesforce user interface.

Step 2: Generate or Obtain the Web Service WSDL

To access the Force.com Web service, you need a Web Service Description Language (WSDL) file. The WSDL file defines the Web service that is available to you. Your development platform uses this WSDL to generate an API to access the Force.com Web service it defines. You can either obtain the WSDL file from your organization's Salesforce administrator or you can generate it yourself if you have access to the WSDL download page in the Salesforce user interface. You can navigate to the most recent WSDL for your organization from Setup by entering "API" in the **Quick Find** box, then selecting **API**.

For more information about WSDL, see <http://www.w3.org/TR/wsdl>.

Generating the WSDL File for Your Organization

Any user with the "Modify All Data" permission can download the Web Services Description Language (WSDL) file to integrate and extend Salesforce using the API. (The System Administrator profile has this permission.)

The WSDL file is dynamically generated based on which type of WSDL file (enterprise or partner) you download. The generated WSDL defines all of the API calls, objects (including standard and custom objects), and fields that are available for API access for your organization.

To generate the WSDL file for your organization:

1. Log in to your Enterprise, Unlimited, Performance, or Developer Edition Salesforce account. You must log in as an administrator or as a user who has the "Modify All Data" permission. Logins are checked to ensure they are from a known IP address. For more information, see "Restrict Where and When Users Can Log In to Salesforce" in the Salesforce online help.
2. From Setup, enter "API" in the **Quick Find** box, then select **API** to display the WSDL download page.
3. Download the **appropriate WSDL**:
 - If you're downloading an enterprise WSDL and you have managed packages installed in your org, click **Generate Enterprise WSDL**. Salesforce prompts you to select the version of each installed package to include in the generated WSDL.
 - Otherwise, right-click the link for the appropriate WSDL document to save it to a local directory. In the menu, Internet Explorer users can choose **Save Target As**, while Mozilla Firefox users can choose **Save Link As**.

Step 3: Import the WSDL File Into Your Development Platform

Once you have the WSDL file, you need to import it into your development platform so that your development environment can generate the necessary objects for use in building client Web service applications in that environment. This section provides sample instructions for WSC and Microsoft Visual Studio. For instructions about other development platforms, see your platform's product documentation.



Note: The process for importing WSDL files is identical for the enterprise and partner WSDL files.

Instructions for Java Environments (WSC)

Java environments access the API through Java objects that serve as proxies for their server-side counterparts. Before using the API, you must first generate these objects from your organization's WSDL file.

Each SOAP client has its own tool for this process. For WSC, use the `wsdlc` utility.



Note: Before you run `wsdlc`, you must have the WSC JAR file installed on your system and referenced in your classpath.

The basic syntax for `wsdlc` is:

```
java -classpath pathToJAR/wsc-22.jar com.sforce.ws.tools.wsdlc pathToWsdl/WsdlFilename  
pathToOutputJar/OutputJarFilename
```

This command generates an output jar file based on the specified WSDL file. After the output jar file is created, reference it along with the wsc jar file (for example, wsc-22.jar) in your Java program to create a client application.

Instructions for Microsoft Visual Studio

Visual Studio languages access the API through objects that serve as proxies for their server-side counterparts. Before using the API, you must first generate these objects from your organization's WSDL file.

Once you have the proxy classes for the server-side objects, you need to ensure that you specify whether you have set any values on non-string fields. For more information, see [Implementation Considerations](#).

Visual Studio provides two approaches for importing your WSDL file and generating an XML Web service client: an IDE-based approach and a command line approach. This walkthrough describes how to import your WSDL file through the IDE.



Note: Before you begin, the first step is to create a new application or open an existing application in Visual Studio. In addition, you need to have generated the WSDL file, as described in [Generating the WSDL File for Your Organization](#).

An XML Web service client is any component or application that references and uses an XML Web service. This does not necessarily need to be a client-based application. In fact, in many cases, your XML Web service clients might be other Web applications, such as Web Forms or even other XML Web services. When accessing XML Web services in managed code, a proxy class and the .NET Framework handle all of the infrastructure coding.

To access an XML Web service from managed code:

1. Name your project `Walkthrough` or change the `using` directive in the following sample to `your_project_name.web_reference_name`. Then, add a Web reference to your project for the XML Web service that you want to access. The Web reference creates a proxy class with methods that serve as proxies for each exposed method of the XML Web service.
2. Add the namespace for the Web reference.
3. Create an instance of the proxy class and then access the methods of that class as you would the methods of any other class.

You can add either a .NET 2.0 style Web reference, or a .NET 3.0 style Service reference, depending on your version of Visual Studio and preferred developer environment. A .NET 3.0 style reference uses services like `SoapClient` instead of `SforceService`.

To add a Web reference:

-  **Note:** These steps may be different depending on the version of Visual Studio that you're using. For more information, see "Adding and Removing Web References" in the Visual Studio documentation.

1. If you are using Visual Studio 2010 or earlier, on the Project menu, choose **Add Web Reference**. For later versions of Visual Studio, on the Project menu, choose **Add Service Reference**, select **Advanced** and then select **Add Web Reference**.
2. In the URL box of the Add Web Reference dialog box, type the URL to obtain the service description of the XML Web service you want to access, such as:
`c:\WSDLFiles\enterprise.wsdl`
3. Click **Go** to retrieve information about the XML Web service.
4. In the Web reference name box, rename the Web reference to `sforce`, which is the name you will use for this Web reference.
5. Click **Add Reference** to add a Web reference for the target XML Web service.
6. Visual Studio retrieves the service description and generates a proxy class to interface between your application and the XML Web service.

-  **Note:** If you are using Visual Basic .Net 1.1 and the enterprise WSDL, you will need to modify the generated Web service client to overcome a bug in Visual Studio's client generation utility. The API exposes two objects (`Case` and `Event`) whose names conflict with Visual Basic keywords. When the classes that represent these objects are created, Visual Studio wraps the class names with brackets (`[Case]` and `[Event]`). This is the method by which you can reuse keywords.

Unfortunately, in the definition of the `sObject` class, Visual Studio does not wrap `Case` and `Event` to class references in the `System.Xml.Serialization.XmlIncludeAttribute` that are part of the `sObject` definition. To work around this problem in Visual Studio, you need to edit the `XmlIncludeAttribute` settings for `Case` and `Event` as shown below. This does not apply to C# and only applies when using the enterprise version of the WSDL.

```
System.Xml.Serialization.XmlIncludeAttribute(GetType([Event])),  
System.Xml.Serialization.XmlIncludeAttribute(GetType([Case])),
```

Step 4: Walk Through the Sample Code

Once you have imported your WSDL file, you can begin building client applications that use the API. Use the following samples to create a basic client application. Comments embedded in the sample explain each section of code.

Java Sample Code

This section walks through a sample Java client application that uses the WSC SOAP client. The purpose of this sample application is to show the required steps for logging into the login server and to demonstrate the invocation and subsequent handling of several API calls.

To run this sample, you must pass the authentication endpoint URL as an argument for your program. You can obtain this URL from the WSDL file. This sample application performs the following main tasks:

1. Prompts the user for their Salesforce username and password.
2. Calls `login()` to log in to the single login server and, if the login succeeds, retrieves user information and writes it to the console along with session information.

3. Calls `describeGlobal()` to retrieve a list of all available objects for the organization's data. The `describeGlobal` method determines the objects that are available to the logged in user. This call should not be made more than once per session, since the data returned from the call is not likely to change frequently. The `DescribeGlobalResult` is echoed to the console.
4. Calls `describeSObject()` to retrieve metadata (field list and object properties) for a specified object. The `describeSObject` method illustrates the type of metadata information that can be obtained for each object available to the user. The sample client application executes a `describeSObjects()` call on the object that the user specifies and then echoes the returned metadata information to the console. Object metadata information includes permissions, field types and lengths, and available values for picklist fields and types for `referenceTo` fields.
5. Calls `query()`, passing a simple query string ("SELECT FirstName, LastName FROM Contact"), and iterating through the returned `QueryResult`.
6. Calls `logout()` to log the user out.

The following sample code uses try/catch blocks to handle exceptions that might be thrown by the API calls.

```
package com.example.samples;

import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.InputStreamReader;
import java.io.IOException;
import com.sforce.soap.enterprise.DeleteResult;
import com.sforce.soap.enterprise.DescribeGlobalResult;
import com.sforce.soap.enterprise.DescribeGlobalSObjectResult;
import com.sforce.soap.enterprise.DescribeSObjectResult;
import com.sforce.soap.enterprise.EnterpriseConnection;
import com.sforce.soap.enterprise.Error;
import com.sforce.soap.enterprise.Field;
import com.sforce.soap.enterprise.FieldType;
import com.sforce.soap.enterprise.GetUserInfoResult;
import com.sforce.soap.enterprise.LoginResult;
import com.sforce.soap.enterprise.PicklistEntry;
import com.sforce.soap.enterprise.QueryResult;
import com.sforce.soap.enterprise.SaveResult;
import com.sforce.soap.enterprise.sobject.Account;
import com.sforce.soap.enterprise.sobject.Contact;
import com.sforce.soap.enterprise.sobject.SObject;
import com.sforce.ws.ConnectorConfig;
import com.sforce.ws.ConnectionException;

public class QuickstartApiSample {

    private static BufferedReader reader = new BufferedReader(
        new InputStreamReader(System.in));

    EnterpriseConnection connection;
    String authEndPoint = "";

    public static void main(String[] args) {
        if (args.length < 1) {
            System.out.println("Usage: com.example.samples."
                + "QuickstartApiSamples <AuthEndPoint>");
            System.exit(-1);
        }
    }
}
```

```
}

QuickstartApiSample sample = new QuickstartApiSample(args[0]);
sample.run();
}

public void run() {
    // Make a login call
    if (login()) {
        // Do a describe global
        describeGlobalSample();

        // Describe an object
        describeSObjectsSample();

        // Retrieve some data using a query
        querySample();

        // Log out
        logout();
    }
}

// Constructor
public QuickstartApiSample(String authEndPoint) {
    this.authEndPoint = authEndPoint;
}

private String getUserInput(String prompt) {
    String result = "";
    try {
        System.out.print(prompt);
        result = reader.readLine();
    } catch (IOException ioe) {
        ioe.printStackTrace();
    }

    return result;
}

private boolean login() {
    boolean success = false;
    String username = getUserInput("Enter username: ");
    String password = getUserInput("Enter password: ");

    try {
        ConnectorConfig config = new ConnectorConfig();
        config.setUsername(username);
        config.setPassword(password);

        System.out.println("AuthEndPoint: " + authEndPoint);
        config.setAuthEndpoint(authEndPoint);

        connection = new EnterpriseConnection(config);
    }
}
```

```
    printUserInfo(config);

    success = true;
} catch (ConnectionException ce) {
    ce.printStackTrace();
}

return success;
}

private void printUserInfo(ConnectorConfig config) {
try {
    GetUserInfoResult userInfo = connection.getUserInfo();

    System.out.println("\nLogging in ...");
    System.out.println("UserID: " + userInfo.getUserId());
    System.out.println("User Full Name: " + userInfo.getUserFullName());
    System.out.println("User Email: " + userInfo.getUserEmail());
    System.out.println();
    System.out.println("SessionID: " + config.getSessionId());
    System.out.println("Auth End Point: " + config.getAuthEndpoint());
    System.out
        .println("Service End Point: " + config.getServiceEndpoint());
    System.out.println();
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

private void logout() {
try {
    connection.logout();
    System.out.println("Logged out.");
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

/**
 * To determine the objects that are available to the logged-in user, the
 * sample client application executes a describeGlobal call, which returns
 * all of the objects that are visible to the logged-in user. This call
 * should not be made more than once per session, as the data returned from
 * the call likely does not change frequently. The DescribeGlobalResult is
 * simply echoed to the console.
*/
private void describeGlobalSample() {
try {
    // describeGlobal() returns an array of object results that
    // includes the object names that are available to the logged-in user.
    DescribeGlobalResult dgr = connection.describeGlobal();

    System.out.println("\nDescribe Global Results:\n");
    // Loop through the array echoing the object names to the console
}
```

```
        for (int i = 0; i < dgr.getObjects().length; i++) {
            System.out.println(dgr.getObjects()[i].getName());
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}

/**
 * The following method illustrates the type of metadata information that can
 * be obtained for each object available to the user. The sample client
 * application executes a describeSObject call on a given object and then
 * echoes the returned metadata information to the console. Object metadata
 * information includes permissions, field types and length and available
 * values for picklist fields and types for referenceTo fields.
*/
private void describeSObjectsSample() {
    String objectToDescribe = getUserInput("\nType the name of the object to "
        + "describe (try Account): ");

    try {
        // Call describeSObjects() passing in an array with one object type
        // name
        DescribeSObjectResult[] dsrArray = connection
            .describeSObjects(new String[] { objectToDescribe });

        // Since we described only one sObject, we should have only
        // one element in the DescribeSObjectResult array.
        DescribeSObjectResult dsr = dsrArray[0];

        // First, get some object properties
        System.out.println("\n\nObject Name: " + dsr.getName());

        if (dsr.getCustom())
            System.out.println("Custom Object");
        if (dsr.getLabel() != null)
            System.out.println("Label: " + dsr.getLabel());

        // Get the permissions on the object

        if (dsr.getCreateable())
            System.out.println("Createable");
        if (dsr.getDeletable())
            System.out.println("Deleteable");
        if (dsr.getQueryable())
            System.out.println("Queryable");
        if (dsr.getReplicable())
            System.out.println("Replicable");
        if (dsr.getRetrievable())
            System.out.println("Retrievable");
        if (dsr.getSearchable())
            System.out.println("Searchable");
        if (dsr.getUndeletable())
            System.out.println("Undeleteable");
    }
}
```

```
if (dsr.getUpdateable())
    System.out.println("Updateable");

System.out.println("Number of fields: " + dsr.getFields().length);

// Now, retrieve metadata for each field
for (int i = 0; i < dsr.getFields().length; i++) {
    // Get the field
    Field field = dsr.getFields()[i];

    // Write some field properties
    System.out.println("Field name: " + field.getName());
    System.out.println("\tField Label: " + field.getLabel());

    // This next property indicates that this
    // field is searched when using
    // the name search group in SOSL
    if (field.getNameField())
        System.out.println("\tThis is a name field.");

    if (field.getRestrictedPicklist())
        System.out.println("This is a RESTRICTED picklist field.");

    System.out.println("\tType is: " + field.getType());

    if (field.getLength() > 0)
        System.out.println("\tLength: " + field.getLength());

    if (field.getScale() > 0)
        System.out.println("\tScale: " + field.getScale());

    if (field.getPrecision() > 0)
        System.out.println("\tPrecision: " + field.getPrecision());

    if (field.getDigits() > 0)
        System.out.println("\tDigits: " + field.getDigits());

    if (field.getCustom())
        System.out.println("\tThis is a custom field.");

    // Write the permissions of this field
    if (field.getNillable())
        System.out.println("\tCan be nulled.");
    if (field.getCreateable())
        System.out.println("\tCreateable");
    if (field.getFilterable())
        System.out.println("\tFilterable");
    if (field.getUpdateable())
        System.out.println("\tUpdateable");

    // If this is a picklist field, show the picklist values
    if (field.getType().equals(FieldType.picklist)) {
        System.out.println("\t\tPicklist values: ");
        PicklistEntry[] picklistValues = field.getPicklistValues();
```

```
        for (int j = 0; j < field.getPicklistValues().length; j++) {
            System.out.println("\t\tValue: "
                + picklistValues[j].getValue());
        }
    }

    // If this is a foreign key field (reference),
    // show the values
    if (field.getType().equals(FieldType.reference)) {
        System.out.println("\t\t\tCan reference these objects:");
        for (int j = 0; j < field.getReferenceTo().length; j++) {
            System.out.println("\t\t\t\t" + field.getReferenceTo()[j]);
        }
    }
    System.out.println("");
}

} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

private void querySample() {
    String soqlQuery = "SELECT FirstName, LastName FROM Contact";
    try {
        QueryResult qr = connection.query(soqlQuery);
        boolean done = false;

        if (qr.getSize() > 0) {
            System.out.println("\nLogged-in user can see "
                + qr.getRecords().length + " contact records.");

            while (!done) {
                System.out.println("");
                SObject[] records = qr.getRecords();
                for (int i = 0; i < records.length; ++i) {
                    Contact con = (Contact) records[i];
                    String fName = con.getFirstName();
                    String lName = con.getLastName();

                    if (fName == null) {
                        System.out.println("Contact " + (i + 1) + ": " + lName);
                    } else {
                        System.out.println("Contact " + (i + 1) + ": " + fName
                            + " " + lName);
                    }
                }
                if (qr.isDone()) {
                    done = true;
                } else {
                    qr = connection.queryMore(qr.getQueryLocator());
                }
            }
        }
    }
}
```

```
        } else {
            System.out.println("No records found.");
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

C# Sample Code

This section walks through a sample C# client application. The purpose of this sample application is to show the required steps for logging in and to demonstrate the invocation and subsequent handling of several API calls.

This sample application performs the following main tasks:

- Prompts the user for their Salesforce username and password.
 - Calls `login()` to log in to the single login server and, if the login succeeds:
 - Sets the returned `sessionId` into the session header, which is required for session authentication on subsequent API calls.
 - Resets the Force.com endpoint to the returned `serverUrl`, which is the target of subsequent API calls.

All client applications that access the API must complete the tasks in this step before attempting any subsequent API calls.
 - Retrieves user information and writes it to the console along with session information.
 - Calls `describeGlobal()` to retrieve a list of all available objects for the organization's data. The `describeGlobal` method determines the objects that are available to the logged in user. This call should not be made more than once per session, since the data returned from the call is not likely to change frequently. The `DescribeGlobalResult` is echoed to the console.
 - Calls `describeSObjects()` to retrieve metadata (field list and object properties) for a specified object. The `describeSObject` method illustrates the type of metadata information that can be obtained for each object available to the user. The sample client application executes a `describeSObjects()` call on the object that the user specifies and then echoes the returned metadata information to the console. Object metadata information includes permissions, field types and lengths, and available values for picklist fields and types for `referenceTo` fields.
 - Calls `query()`, passing a simple query string ("SELECT FirstName, LastName FROM Contact"), and iterating through the returned `QueryResult`.
 - Calls `logout()` to log the user out.

The following sample code uses try/catch blocks to handle exceptions that might be thrown by the API calls.

The following code begins the sample C# client application.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web.Services.Protocols;
using Walkthrough.sforce;

namespace Walkthrough
{
    class QuickstartApiSample
    {
}
```

```
private SforceService binding;

[STAThread]
static void Main(string[] args)
{
    QuickstartApiSample sample = new QuickstartApiSample();
    sample.run();
}

public void run()
{
    // Make a login call
    if (login())
    {
        // Do a describe global
        describeGlobalSample();

        // Describe an account object
        describeSObjectsSample();

        // Retrieve some data using a query
        querySample();

        // Log out
        logout();
    }
}

private bool login()
{
    Console.WriteLine("Enter username: ");
    string username = Console.ReadLine();
    Console.WriteLine("Enter password: ");
    string password = Console.ReadLine();

    // Create a service object
    binding = new SforceService();

    // Timeout after a minute
    binding.Timeout = 60000;

    // Try logging in
    LoginResult lr;
    try
    {

        Console.WriteLine("\nLogging in...\n");
        lr = binding.login(username, password);
    }

    // ApiFault is a proxy stub generated from the WSDL contract when
    // the web service was imported
    catch (SoapException e)
    {
```

```
// Write the fault code to the console
Console.WriteLine(e.Code);

// Write the fault message to the console
Console.WriteLine("An unexpected error has occurred: " + e.Message);

// Write the stack trace to the console
Console.WriteLine(e.StackTrace);

// Return False to indicate that the login was not successful
return false;
}

// Check if the password has expired
if (lr.passwordExpired)
{
    Console.WriteLine("An error has occurred. Your password has expired.");
    return false;
}

/** Once the client application has logged in successfully, it will use
 * the results of the login call to reset the endpoint of the service
 * to the virtual server instance that is servicing your organization
 */
// Save old authentication end point URL
String authEndPoint = binding.Url;
// Set returned service endpoint URL
binding.Url = lr.serverUrl;

/** The sample client application now has an instance of the SforceService
 * that is pointing to the correct endpoint. Next, the sample client
 * application sets a persistent SOAP header (to be included on all
 * subsequent calls that are made with SforceService) that contains the
 * valid sessionId for our login credentials. To do this, the sample
 * client application creates a new SessionHeader object and persist it to
 * the SforceService. Add the session ID returned from the login to the
 * session header
 */
binding.SessionHeaderValue = new SessionHeader();
binding.SessionHeaderValue.sessionId = lr.sessionId;

printUserInfo(lr, authEndPoint);

// Return true to indicate that we are logged in, pointed
// at the right URL and have our security token in place.
return true;
}

private void printUserInfo(LoginResult lr, String authEP)
{
    try
```

```
{  
    GetUserUserInfoResult userInfo = lr.userInfo;  
  
    Console.WriteLine("\nLogging in ...");  
    Console.WriteLine("UserID: " + userInfo.userId);  
    Console.WriteLine("User Full Name: " +  
        userInfo.userFullName);  
    Console.WriteLine("User Email: " +  
        userInfo.userEmail);  
    Console.WriteLine();  
    Console.WriteLine("SessionID: " +  
        lr.sessionId);  
    Console.WriteLine("Auth End Point: " +  
        authEP);  
    Console.WriteLine("Service End Point: " +  
        lr.serverUrl);  
    Console.WriteLine();  
}  
}  
  
catch (SoapException e)  
{  
    Console.WriteLine("An unexpected error has occurred: " + e.Message +  
        " Stack trace: " + e.StackTrace);  
}  
}  
}  
  
private void logout()  
{  
    try  
    {  
        binding.logout();  
        Console.WriteLine("Logged out.");  
    }  
    catch (SoapException e)  
    {  
        // Write the fault code to the console  
        Console.WriteLine(e.Code);  
  
        // Write the fault message to the console  
        Console.WriteLine("An unexpected error has occurred: " + e.Message);  
  
        // Write the stack trace to the console  
        Console.WriteLine(e.StackTrace);  
    }  
}  
}  
  
/**  
 * To determine the objects that are available to the logged-in  
 * user, the sample client application executes a describeGlobal  
 * call, which returns all of the objects that are visible to  
 * the logged-in user. This call should not be made more than  
 * once per session, as the data returned from the call likely  
 * does not change frequently. The DescribeGlobalResult is  
 * simply echoed to the console.  
 */
```

```
private void describeGlobalSample()
{
    try
    {
        // describeGlobal() returns an array of object results that
        // includes the object names that are available to the logged-in user.
        DescribeGlobalResult dgr = binding.describeGlobal();

        Console.WriteLine("\nDescribe Global Results:\n");
        // Loop through the array echoing the object names to the console

        for (int i = 0; i < dgr.sobjects.Length; i++)
        {
            Console.WriteLine(dgr.sobjects[i].name);
        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An exception has occurred: " + e.Message +
            "\nStack trace: " + e.StackTrace);
    }
}

/**
 * The following method illustrates the type of metadata
 * information that can be obtained for each object available
 * to the user. The sample client application executes a
 * describeSObject call on a given object and then echoes
 * the returned metadata information to the console. Object
 * metadata information includes permissions, field types
 * and length and available values for picklist fields
 * and types for referenceTo fields.
 */
private void describeSObjectsSample()
{
    Console.Write("\nType the name of the object to " +
        "describe (try Account): ");
    string objectType = Console.ReadLine();
    try
    {

        // Call describeSObjects() passing in an array with one object type name
        DescribeSObjectResult[] dsrArray =
            binding.describeSObjects(new string[] { objectType });

        // Since we described only one sObject, we should have only
        // one element in the DescribeSObjectResult array.
        DescribeSObjectResult dsr = dsrArray[0];

        // First, get some object properties
        Console.WriteLine("\n\nObject Name: " + dsr.name);

        if (dsr.custom) Console.WriteLine("Custom Object");
        if (dsr.label != null) Console.WriteLine("Label: " + dsr.label);
    }
}
```

```
// Get the permissions on the object
if (dsr.createable) Console.WriteLine("Createable");
if (dsr.deletable) Console.WriteLine("Deleteable");
if (dsr.queryable) Console.WriteLine("Queryable");
if (dsr.replicateable) Console.WriteLine("Replicable");
if (dsr.retrieveable) Console.WriteLine("Retrievable");
if (dsr.searchable) Console.WriteLine("Searchable");
if (dsr.undeletable) Console.WriteLine("Undeleteable");
if (dsr.updateable) Console.WriteLine("Updateable");

Console.WriteLine("Number of fields: " + dsr.fields.Length);

// Now, retrieve metadata for each field
for (int i = 0; i < dsr.fields.Length; i++)
{
    // Get the field
    Field field = dsr.fields[i];

    // Write some field properties
    Console.WriteLine("Field name: " + field.name);
    Console.WriteLine("\tField Label: " + field.label);

    // This next property indicates that this
    // field is searched when using
    // the name search group in SOSL
    if (field.nameField)
        Console.WriteLine("\tThis is a name field.");

    if (field.restrictedPicklist)
        Console.WriteLine("This is a RESTRICTED picklist field.");

    Console.WriteLine("\tType is: " + field.type.ToString());

    if (field.length > 0)
        Console.WriteLine("\tLength: " + field.length);

    if (field.scale > 0)
        Console.WriteLine("\tScale: " + field.scale);

    if (field.precision > 0)
        Console.WriteLine("\tPrecision: " + field.precision);

    if (field.digits > 0)
        Console.WriteLine("\tDigits: " + field.digits);

    if (field.custom)
        Console.WriteLine("\tThis is a custom field.");

    // Write the permissions of this field
    if (field.nillable) Console.WriteLine("\tCan be nulled.");
    if (field.createable) Console.WriteLine("\tCreateable");
    if (field.filterable) Console.WriteLine("\tFilterable");
    if (field.updateable) Console.WriteLine("\tUpdateable");
```

```

// If this is a picklist field, show the picklist values
if (field.type.Equals(fieldType.picklist))
{
    Console.WriteLine("\tPicklist Values");
    for (int j = 0; j < field.picklistValues.Length; j++)
        Console.WriteLine("\t\t" + field.picklistValues[j].value);
}

// If this is a foreign key field (reference),
// show the values
if (field.type.Equals(fieldType.reference))
{
    Console.WriteLine("\tCan reference these objects:");
    for (int j = 0; j < field.referenceTo.Length; j++)
        Console.WriteLine("\t\t" + field.referenceTo[j]);
}
Console.WriteLine("");
}

catch (SoapException e)
{
    Console.WriteLine("An exception has occurred: " + e.Message +
"\nStack trace: " + e.StackTrace);
}
Console.WriteLine("Press ENTER to continue...");
Console.ReadLine();
}

private void querySample()
{
    String soqlQuery = "SELECT FirstName, LastName FROM Contact";
    try
    {
        QueryResult qr = binding.query(soqlQuery);
        bool done = false;

        if (qr.size > 0)
        {
            Console.WriteLine("Logged-in user can see "
                + qr.records.Length + " contact records.");

            while (!done)
            {
                Console.WriteLine("");
                sObject[] records = qr.records;
                for (int i = 0; i < records.Length; i++)
                {
                    Contact con = (Contact)records[i];
                    string fName = con.FirstName;
                    string lName = con.LastName;
                    if (fName == null)
                        Console.WriteLine("Contact " + (i + 1) + ": " + lName);
                    else

```

```
        Console.WriteLine("Contact " + (i + 1) + ": " + fName
                           + " " + lName);
    }

    if (qr.done)
    {
        done = true;
    }
    else
    {
        qr = binding.queryMore(qr.queryLocator);
    }
}

else
{
    Console.WriteLine("No records found.");
}

}

catch (Exception ex)
{
    Console.WriteLine("\nFailed to execute query successfully," +
                     "error message was: \n{0}", ex.Message);
}

Console.WriteLine("\nPress ENTER to continue...");  

Console.ReadLine();
}
}
```

The following C# example is the same as the previous C# example, except it uses .NET 3.0 SoapClient services instead of .NET 2.0 SforceService services.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using System.ServiceModel;
using Walkthrough.sforce;

namespace Walkthrough
{
    class QuickstartApiSample
    {
        private static SoapClient loginClient; // for login endpoint
        private static SoapClient client; // for API endpoint
        private static SessionHeader header;
        private static EndpointAddress endpoint;

        static void Main(string[] args)
        {
            QuickstartApiSample sample = new QuickstartApiSample();
            sample.run();
        }
    }
}
```

```
}

public void run()
{
    // Make a login call
    if (login())
    {
        // Do a describe global
        describeGlobalSample();

        // Describe an account object
        describeSObjectsSample();

        // Retrieve some data using a query
        querySample();

        // Log out
        logout();
    }
}

private bool login()
{
    Console.Write("Enter username: ");
    string username = Console.ReadLine();
    Console.Write("Enter password: ");
    string password = Console.ReadLine();

    // Create a SoapClient specifically for logging in
    loginClient = new SoapClient();

    // (combine pw and token if necessary)
    LoginResult lr;
    try
    {
        Console.WriteLine("\nLogging in...\n");
        lr = loginClient.login(null, username, password);
    }
    catch (Exception e)
    {
        // Write the fault message to the console
        Console.WriteLine("An unexpected error has occurred: " + e.Message);

        // Write the stack trace to the console
        Console.WriteLine(e.StackTrace);
        return false;
    }

    // Check if the password has expired
    if (lr.passwordExpired)
    {
        Console.WriteLine("An error has occurred. Your password has expired.");
        return false;
    }
}
```

```
/** Once the client application has logged in successfully, it will use
 * the results of the login call to reset the endpoint of the service
 * to the virtual server instance that is servicing your organization
 */

// On successful login, cache session info and API endpoint info
endpoint = new EndpointAddress(lr.serverUrl);

/** The sample client application now has a cached EndpointAddress
 * that is pointing to the correct endpoint. Next, the sample client
 * application sets a persistent SOAP header that contains the
 * valid sessionId for our login credentials. To do this, the sample
 * client application creates a new SessionHeader object. Add the session
 * ID returned from the login to the session header
 */
header = new SessionHeader();
header.sessionId = lr.sessionId;

// Create and cache an API endpoint client
client = new SoapClient("Soap", endpoint);

printUserInfo(lr, lr.serverUrl);

// Return true to indicate that we are logged in, pointed
// at the right URL and have our security token in place.
return true;
}

private void printUserInfo(LoginResult lr, String authEP)
{
    try
    {
        GetUserInfoResult userInfo = lr.userInfo;

        Console.WriteLine("\nLogging in ...");
        Console.WriteLine("UserID: " + userInfo.userId);
        Console.WriteLine("User Full Name: " +
            userInfo.userFullName);
        Console.WriteLine("User Email: " +
            userInfo.userEmail);
        Console.WriteLine();
        Console.WriteLine("SessionID: " +
            lr.sessionId);
        Console.WriteLine("Auth End Point: " +
            authEP);
        Console.WriteLine("Service End Point: " +
            lr.serverUrl);
        Console.WriteLine();
    }
    catch (Exception e)
    {
        Console.WriteLine("An unexpected error has occurred: " + e.Message +
            " Stack trace: " + e.StackTrace);
    }
}
```

```
        }

    }

    private void logout()
    {
        try
        {
            client.logout(header);
            Console.WriteLine("Logged out.");
        }
        catch (Exception e)
        {
            // Write the fault message to the console
            Console.WriteLine("An unexpected error has occurred: " + e.Message);

            // Write the stack trace to the console
            Console.WriteLine(e.StackTrace);
        }
    }

    /**
     * To determine the objects that are available to the logged-in
     * user, the sample client application executes a describeGlobal
     * call, which returns all of the objects that are visible to
     * the logged-in user. This call should not be made more than
     * once per session, as the data returned from the call likely
     * does not change frequently. The DescribeGlobalResult is
     * simply echoed to the console.
     */
    private void describeGlobalSample()
    {
        try
        {
            // describeGlobal() returns an array of object results that
            // includes the object names that are available to the logged-in user.
            DescribeGlobalResult dgr = client.describeGlobal(
                header, // session header
                null // package version header
            );

            Console.WriteLine("\nDescribe Global Results:\n");
            // Loop through the array echoing the object names to the console

            for (int i = 0; i < dgr.sobjects.Length; i++)
            {
                Console.WriteLine(dgr.sobjects[i].name);
            }
        }
        catch (Exception e)
        {
            Console.WriteLine("An exception has occurred: " + e.Message +
                "\nStack trace: " + e.StackTrace);
        }
    }
}
```

```
/**  
 * The following method illustrates the type of metadata  
 * information that can be obtained for each object available  
 * to the user. The sample client application executes a  
 * describeSObject call on a given object and then echoes  
 * the returned metadata information to the console. Object  
 * metadata information includes permissions, field types  
 * and length and available values for picklist fields  
 * and types for referenceTo fields.  
 */  
private void describeSObjectsSample()  
{  
    Console.WriteLine("\nType the name of the object to " +  
        "describe (try Account): ");  
    string objectType = Console.ReadLine();  
    try  
    {  
  
        // Call describeSObjects() passing in an array with one object type name  
  
        DescribeSObjectResult[] dsrArray =  
            client.describeSObjects(  
                header, // session header  
                null, // package version header  
                null, // locale options  
                new string[] { objectType } // object name array  
            );  
  
        // Since we described only one sObject, we should have only  
        // one element in the DescribeSObjectResult array.  
        DescribeSObjectResult dsr = dsrArray[0];  
  
        // First, get some object properties  
        Console.WriteLine("\n\nObject Name: " + dsr.name);  
  
        if (dsr.custom) Console.WriteLine("Custom Object");  
        if (dsr.label != null) Console.WriteLine("Label: " + dsr.label);  
  
        // Get the permissions on the object  
        if (dsr.createable) Console.WriteLine("Createable");  
        if (dsr.deletable) Console.WriteLine("Deleteable");  
        if (dsr.queryable) Console.WriteLine("Queryable");  
        if (dsr.replicable) Console.WriteLine("Replicable");  
        if (dsr.retrieveable) Console.WriteLine("Retrieveable");  
        if (dsr.searchable) Console.WriteLine("Searchable");  
        if (dsr.undeletable) Console.WriteLine("Undeleteable");  
        if (dsr.updateable) Console.WriteLine("Updateable");  
  
        Console.WriteLine("Number of fields: " + dsr.fields.Length);  
  
        // Now, retrieve metadata for each field  
        for (int i = 0; i < dsr.fields.Length; i++)  
    }
```

```
// Get the field
Field field = dsr.fields[i];

// Write some field properties
Console.WriteLine("Field name: " + field.name);
Console.WriteLine("\tField Label: " + field.label);

// This next property indicates that this
// field is searched when using
// the name search group in SOSL
if (field.nameField)
    Console.WriteLine("\tThis is a name field.");

if (field.restrictedPicklist)
    Console.WriteLine("This is a RESTRICTED picklist field.");

Console.WriteLine("\tType is: " + field.type.ToString());

if (field.length > 0)
    Console.WriteLine("\tLength: " + field.length);

if (field.scale > 0)
    Console.WriteLine("\tScale: " + field.scale);

if (field.precision > 0)
    Console.WriteLine("\tPrecision: " + field.precision);

if (field.digits > 0)
    Console.WriteLine("\tDigits: " + field.digits);

if (field.custom)
    Console.WriteLine("\tThis is a custom field.");

// Write the permissions of this field
if (field.nullable) Console.WriteLine("\tCan be nulled.");
if (field.createable) Console.WriteLine("\tCreateable");
if (field.filterable) Console.WriteLine("\tFilterable");
if (field.updateable) Console.WriteLine("\tUpdateable");

// If this is a picklist field, show the picklist values
if (field.type.Equals(fieldType.picklist))
{
    Console.WriteLine("\tPicklist Values");
    for (int j = 0; j < field.picklistValues.Length; j++)
        Console.WriteLine("\t\t" + field.picklistValues[j].value);
}

// If this is a foreign key field (reference),
// show the values
if (field.type.Equals(fieldType.reference))
{
    Console.WriteLine("\tCan reference these objects:");
    for (int j = 0; j < field.referenceTo.Length; j++)
        Console.WriteLine("\t\t" + field.referenceTo[j]);
```

```
        }
        Console.WriteLine("");
    }
}
catch (Exception e)
{
    Console.WriteLine("An exception has occurred: " + e.Message +
        "\nStack trace: " + e.StackTrace);
}
Console.WriteLine("Press ENTER to continue...");  
Console.ReadLine();
}

private void querySample()
{
    String soqlQuery = "SELECT FirstName, LastName FROM Contact";
    try
    {
        QueryResult qr = client.query(
            header, // session header
            null, // query options
            null, // mru options
            null, // package version header
            soqlQuery // query string
        );

        bool done = false;

        if (qr.size > 0)
        {
            Console.WriteLine("Logged-in user can see "
                + qr.records.Length + " contact records.");

            while (!done)
            {
                Console.WriteLine("");
                sObject[] records = qr.records;
                for (int i = 0; i < records.Length; i++)
                {
                    Contact con = (Contact)records[i];
                    string fName = con.FirstName;
                    string lName = con.LastName;
                    if (fName == null)
                        Console.WriteLine("Contact " + (i + 1) + ": " + lName);
                    else
                        Console.WriteLine("Contact " + (i + 1) + ": " + fName
                            + " " + lName);
                }

                if (qr.done)
                {
                    done = true;
                }
            }
        }
    }
}
```

```
        {
            qr = client.queryMore(
                header, // session header
                null, // query options
                qr.queryLocator // query locator
            );
        }
    }
}
else
{
    Console.WriteLine("No records found.");
}
}
catch (Exception ex)
{
    Console.WriteLine("\nFailed to execute query successfully," +
        "error message was: \n{0}", ex.Message);
}
Console.WriteLine("\nPress ENTER to continue...");  
Console.ReadLine();
}
}
}
```

CHAPTER 2 Object Basics

In this chapter ...

- [Primitive Data Types](#)
- [Field Types](#)
- [Compound Fields](#)
- [API Data Types and Salesforce Field Types](#)
- [Core Data Types Used in API Calls](#)
- [System Fields](#)
- [Required Fields](#)
- [Frequently-Occurring Fields](#)
- [API Field Properties](#)
- [Relationships Among Objects](#)
- [Relabeling Fields and Tabs and the API](#)
- [Tooling API Objects in the Enterprise WSDL](#)
- [Force.com AppExchange Object Prefixes and the API](#)
- [Custom Objects](#)
- [External Objects](#)

Generally speaking, API objects represent database tables that contain your organization's information. For example, the central object in the Salesforce data model represents accounts—companies and organizations involved with your business, such as customers, partners, and competitors. The term "record" describes a particular occurrence of an object (such as a specific account like "IBM" or "United Airlines" that is represented by an Account object). A record is analogous to a row in a database table.

Objects already created for you by Salesforce are called standard objects. Objects you create in your organization are called custom objects. Objects you create that map to data stored outside your organization are called external objects.

While this document describes all of the objects available in the API, your applications work with only the objects that you are authorized to access. Programmatic access to objects is determined by the objects defined in your organization, your organization configuration, your user permissions and access settings (which are configured by your organization's system administrator), your data sharing model, and other factors related specifically to the object.

Most of the objects accessible through the API are read-write objects. However, there are a few objects that are read-only. This fact is noted in the description for the object.

Primitive Data Types

The API uses the following primitive data types:

Value	Details
base64	Base 64-encoded binary data. Fields of this type are used for storing binary files in Attachment records, Document records, and Scontrol records. In these objects, the <code>Body</code> or <code>Binary</code> field contains the (base64 encoded) data, while the <code>BodyLength</code> field defines the length of the data in the <code>Body</code> or <code>Binary</code> field. In the Document object, you can specify a URL to the document instead of storing the document directly in the record.
boolean	Boolean fields have one of these values: <code>true</code> (or 1), or <code>false</code> (or 0).
byte	A set of bits.
date	Date data. Fields of this type contain date values, such as <code>ActivityDate</code> in the Event object. Unlike <code>dateTime</code> fields, date fields contain no time value—the time portion of a date field is not relevant and is always set to midnight in the Coordinated Universal Time (UTC) time zone. If you specify a date value in a query, you can filter on date fields only.
dateTime	Date/time values (timestamps). Fields of this type handle date/time values (timestamps), such as <code>ActivityDateTime</code> in the Event object or the <code>CreatedDate</code> , <code>LastModifiedDate</code> , or <code>SystemModstamp</code> in many objects. Regular <code>dateTime</code> fields are full timestamps with a precision of one second. They are always transferred in the Coordinated Universal Time (UTC) time zone. In your client application, you might need to translate the timestamp to or from a local time zone. If you specify a <code>dateTime</code> value in a query, you can filter on <code>dateTime</code> fields only. Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time zone. To determine how your development tool handles time values, refer to its documentation.
<p> Note: The Event object has a <code>DurationInMinutes</code> field that specifies the number of minutes for an event. Even though this is a temporal value, it is an integer type—not a <code>dateTime</code> type.</p>	
double	Double values. Fields of this type can contain fractional portions (digits to the right of the decimal place), such as <code>ConversionRate</code> in <code>CurrencyType</code> . In the API, all non-integer values (such as <code>Currency Field Type</code> and <code>Percent Field Type</code>) contain values of type double. Some restrictions may be applied to double values: <ul style="list-style-type: none"> • <code>scale</code>: Maximum number of digits to the right of the decimal place. • <code>precision</code>: Total number of digits, including those to the left and the right of the decimal place The maximum number of digits to the left of the decimal place is equal to <code>precision</code> minus <code>scale</code> . In the online application, precision is defined differently—it is the maximum number of digits allowed to the left of the decimal place. Values can be stored in scientific notation if the number is large enough (or, for negative numbers, small enough), as indicated by the W3C XML Schema Part 2: Datatypes Second Edition specification .
<p> Warning: When the user sets the precision in custom fields in the Salesforce application, it displays the precision set by the user, even if the user enters a more precise value than defined for those</p>	

Value	Details
	fields. However, when you set the precision in custom fields using the API, no rounding occurs when the user retrieves the number field.
int	Fields of this type contain numbers with no fractional portion (digits to the right of a decimal place), such as the <code>NumberOfEmployees</code> in an Account. For integer fields, the digits field specifies the maximum number of digits that an integer can have.
string	Character strings. Fields that are of data type <code>string</code> contain text and some have length restrictions depending on the data being stored. For example, in the Contact object, the <code>FirstName</code> field is 40 characters, the <code>LastName</code> field is 80 characters, the <code>MailingStreet</code> is 255 characters.
	<p> Note: For fields that contain strings, behavior is different beginning with API version 15.0. In API versions previous to 15.0, if you specify a value for a field, and that value is too large, the value is truncated. For API version 15.0 and later, if a value is specified that is too large, the operation fails and the fault code <code>STRING_TOO_LONG</code> is returned. <code>AllowFieldTruncationHeader</code> allows you to specify that the previous behavior, truncation, be used instead of the new behavior in API versions 15.0 and later. This header has no effect in versions 14.0 and earlier. The affected fields are: <code>anyType</code>, <code>email</code>, <code>encryptedstring</code>, <code>multipicklist</code>, <code>phone</code>, <code>picklist</code>, <code>string</code>, and <code>textarea</code>.</p>
time	Time values. Fields of this type handle time values, such as <code>FridayEndTime</code> in the <code>BusinessHours</code> object. Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time zone. To determine how your development tool handles time values, refer to its documentation.

These data types are used in the SOAP messages that are exchanged between your client application and the API. When writing your client application, follow the data typing rules defined for your programming language and development environment. Your development tool handles the mapping of typed data in your programming language with these SOAP data types.

The primitive data types are:

- specified in the World Wide Web Consortium's publication *XML Schema Part 2: Data Types* at the following URL:
<http://www.w3.org/TR/xmlschema-2/>.
- enumerated in the `SOAPType` field of the Field type, which is described in the `fields` property of the `DescribeSObjectResult`.

Primitive types are used as a standardized way to define, send, receive, and interpret basic data types in the SOAP messages exchanged between client applications and the API. In addition, primitive data types are interpreted in a Salesforce-specific way, which is useful for display formatting and for numeric conversion (adding values of different currencies).

For example, Salesforce chooses to interpret a double value passed via SOAP as a `double` in a number of possible ways, depending on the field definition. If the field type for that data is currency, Salesforce handles the display of the data by prepending it with a currency symbol and inserting a decimal for precision. Similarly, if the field type is percent, Salesforce handles the display of the data by appending a percent sign (%). Regardless of the field type, however, the value is sent in the SOAP message as a double.

The API uses data types called field types that are defined in the WSDLs. For more information, see [Field Types](#).

Field Types

In addition to the primitive data types, the API defines the following data types for fields:

 **Note:** For fields that contain strings, behavior is different beginning with API version 15.0. In API versions previous to 15.0, if you specify a value for a field, and that value is too large, the value is truncated. For API version 15.0 and later, if a value is specified that is too large, the operation fails and the fault code `STRING_TOO_LONG` is returned. `AllowFieldTruncationHeader` allows you to specify that the previous behavior, truncation, be used instead of the new behavior in API versions 15.0 and later. This header has no effect in versions 14.0 and earlier. The affected fields are: `anyType`, `email`, `encryptedstring`, `multipicklist`, `phone`, `picklist`, `string`, and `textarea`.

Field Type	What the Field Contains
address	A compound data type that contains address field data. See Address Compound Fields .
anyType	Polymorphic data type that returns string, picklist, reference, Boolean, currency, int, double, percent, ID, date, datetime, url, or email data depending on the kind of field involved. See AnyType Field Type .
calculated	Fields that are defined by a formula. See Calculated Field Type .
combobox	A combobox, which includes a set of enumerated values and allows the user to specify a value not in the list. See ComboBox Field Type .
currency	Currency values. See Currency Field Type .
DataCategoryGroupReference	Reference to a data category group or a category unique name. See DataCategoryGroupReference Field Type .
email	Email addresses. See Email Field Type .
encryptedstring	Encrypted text fields contain any combination of letters, numbers, or symbols that are stored in encrypted form. You can set a maximum length of up to 175 characters. Available in API versions 11.0 and later.
ID	Primary key field for the object. See ID Field Type . Note that most Web services tools, including .NET and WSC, map the ID simple type defined in the API WSDL (Enterprise or Partner) to a string. However, other tools generate a specific ID class to represent the ID simple type. Please consult your Web services toolkit documentation for more information.
JunctionIdList	A string array of referenced ID values that represent the many-to-many relationship of an underlying junction entity. Query and manipulate the string array to query and manipulate the underlying junction entities in a single API call. See: JunctionIdList Field Type  Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.
location	A compound data type that contains latitude and longitude values for geolocation fields. See Geolocation Compound Field .
masterrecord	When records are merged, the ID of the record that is saved (the other records are deleted).
multipicklist	Multi-select picklists, which include a set of enumerated values from which multiple values can be selected. See Multi-Select Picklist Field Type .
percent	Percentage values. See Percent Field Type .

Field Type	What the Field Contains
phone	Phone numbers. Values can include alphabetic characters. Client applications are responsible for phone number formatting. See Phone Field Type .
picklist	Picklists, which include a set of enumerated values from which one value can be selected. See Picklist Field Type .
reference	Cross-references to a different object. Analogous to a foreign key field in SQL. See Reference Field Type .
textarea	String that is displayed as a multiline text field. See Textarea Field Type .
url	URL values. Client applications should commonly display these as hyperlinks. See URL Field Type .

These field types extend [primitive data types](#). While many of these field types follow common data typing conventions that are made explicit in their metadata, certain field types have unique characteristics that you need to understand before using them in your client application.

These field types apply to both standard and custom fields. They are enumerated in the `type` field of the [Field](#) type, which is described in the `fields` property of the `DescribeSObjectResult`.

 **Note:** Some numeric fields have precision and scale limits. In addition, certain text fields have length restrictions. These restrictions are enforced when you `create()` or `update()` objects. However, the API may return data that does not meet these restrictions.

AnyType Field Type

The `anyType` field type is dynamic and returns `string`, `date`, `number`, or `boolean` data depending on the kind of field involved. For example, the element in a SOAP message has an `xsi:type="xsd:string"` attribute if the field is of type `string`. This field type is used in history objects for the `NewValue` and `OldValue` fields. It is also a valid field type for `fieldType` and `soapType`.

 **Note:** Most SOAP toolkits automatically deserialize this element into the correct native type.

Calculated Field Type

Calculated fields are read-only fields in the API. These are fields defined by a formula, which is an algorithm that derives its value from other fields, expressions, or values. You can filter on these fields in SOQL, but you should not replicate these fields. The length of text calculated fields is 3900 characters or less—anything longer will be truncated.

Calculated fields are called formula fields in the Salesforce user interface.

ComboBox Field Type

A combobox is a picklist that also allows users to type a value that is not already specified in the list. A combobox is defined as a string value.

Currency Field Type

Currency fields contain currency values, such as the `ExpectedRevenue` field in a Campaign, and are defined as type double.

For organizations that have the multicurrency option enabled, the `CurrencyIsoCode` field is defined for any object that can have currency fields. The `CurrencyIsoCode` field and currency fields are linked in a special way. On any specific record, the `CurrencyIsoCode` field defines the currency of that record, and thus, the values of all currency fields on that record will be expressed in that currency.

For most cases, clients do not need to consider the linking of the `CurrencyIsoCode` field and the currency fields on an object. However, clients may need to consider the following:

- The `CurrencyIsoCode` field exists only for those organizations that have enabled multicurrency support.
- When displaying the currency values in a user interface, it is preferred to prepend each currency value with its `CurrencyIsoCode` value and a space separator.
- The `CurrencyIsoCode` field is a restricted picklist field. The set of allowable values, defined in the `CurrencyType` object, can vary from organization to organization. Attempting to set it to a value that is not defined for an organization causes the operation to be rejected.
- If you update the `CurrencyIsoCode` field on an object, it implicitly converts all currency values on that object to the new currency code, using the conversion rates that are defined for that organization in the Salesforce user interface. If you specify currency values in that same `update()` call, the new currency values you specify are interpreted in the new `CurrencyIsoCode` field value, without conversion.
- The picklist values in a `CurrencyIsoCode` field do not exactly match the labels displayed in Salesforce.

To perform currency conversions, client applications can look up the `CurrencyIsoCode` in the `CurrencyType` object.

DataCategoryGroupReference Field Type

A data category group has categories that classify articles in Salesforce Knowledge and questions in the Answers feature. Every article and question object has two fields of type `DataCategoryGroupReference` which contain the category group and category unique name. You can use the `describeDataCategoryGroups()` and `describeDataCategoryGroupStructures()` calls to retrieve the category groups and categories associated to these objects.

Email Field Type

Email fields contain email addresses. Client applications are responsible for specifying valid and properly formatted email addresses in `create()` and `update()` calls.

ID Field Type

With rare exceptions, all objects in the API have a field of type `ID` that is named `Id` and contains a unique identifier for each record in the object. It is analogous to a primary key in relational databases. When you `create()` a new record, the Web service generates an `ID` value for the record, ensuring that it is unique within your organization's data. You cannot use the `update()` call on `ID` fields. Because the `ID` value stays constant over the lifetime of the record, you can refer to the record by its `ID` value in subsequent API calls. Also, the `ID` value contains a three-character code that identifies the object type, which client applications can retrieve via the `describeSObjects()` call.

In addition, certain objects, including custom objects, have one or more fields of type `reference` that contain the `ID` value for a related record. These fields have names that end in the suffix “-Id”, for example, `OwnerId` in the account object. `OwnerId` contains the `ID` of the user who owns that object. Unlike the field named `Id`, `reference` fields are analogous to foreign keys and can be changed via the `update()` call. For more information, see [Reference Field Type](#).

Some API calls, such as `retrieve()` and `delete()`, accept an array of IDs as parameters—each array element uniquely identifies the row to retrieve or delete. Similarly, the `update()` call accepts an array of sObject records—each sObject contains an `ID` field that uniquely identifies the sObject.

ID fields in the Salesforce user interface contain 15-character, base-62, case-sensitive strings. Each of the 15 characters can be a numeric digit (0-9), a lowercase letter (a-z), or an uppercase letter (A-Z). Two unique IDs may only be different by a change in case.

Because there are applications like Access which do not recognize that 50130000000014c is a different ID from 50130000000014C, an 18-digit, case-safe version of the ID is returned by all API calls. The 18 character IDs have been formed by adding a suffix to each ID in the Force.com API. 18-character IDs can be safely compared for uniqueness by case-insensitive applications, and can be used in all API calls when creating, editing, or deleting data.

If you need to convert the 18-character ID to a 15-character version, truncate the last three characters. Salesforce recommends that you use the 18-character ID.



Note: Most Web services tools, including .NET and WSC, map the ID simple type defined in the API WSDL (Enterprise or Partner) to a string. However, other tools generate a specific ID class to represent the ID simple type. Please consult your web services toolkit documentation for more information.

JunctionIdList Field Type

Starting in API version 34.0, the `JunctionIdList` field type lets you manipulate the many-to-many relationship of an entity directly. You no longer need to manipulate underlying junction entity records. `JunctionIdList` fields can be queried and updated like any other field on the entity. Queries or updates to `JunctionIdList` fields act as queries or updates to the underlying junction entity records. Fields of type `JunctionIdList` appear in the WSDL as an unbounded array of type `ID`.

Multi-Select Picklist Field Type

Multi-select picklist fields contain a list of one or more items from which a user can choose multiple items. One of the items can be configured as the default item. Selections are maintained as a string containing a series of attributes delimited by semicolons. For example, a query might return the values of a multivalue picklist as “first value; second value; third value”. For information on querying multi-select picklists, see Querying Multi-Select Picklists in the [Salesforce SOQL and SOSL Reference Guide](#).

Percent Field Type

Percent fields contain percent values. Percent fields are defined as type `double`.

Phone Field Type

Phone fields contain phone numbers, which can include alphabetic characters. Client applications are responsible for phone number formatting.

Picklist Field Type

Picklist fields contain a list of one or more items from which a user chooses a single item. They display as drop-down lists in the Salesforce user interface. One of the items can be configured as the default item.

In the `Field` object associated with the `DescribeSObjectResult`, the `restrictedPicklist` field defines whether the field is a restricted picklist or not. The API does not enforce the list of values for advisory (unrestricted) picklist fields on `create()` or `update()`. When inserting an unrestricted picklist field that does not have a `PicklistEntry`, the system creates an “inactive” picklist value. This value can be promoted to an “active” picklist value by adding the picklist value in the Salesforce user interface.

When creating new, inactive picklists, the API checks to see if there is a match. This check is case-insensitive.

In the `Field` object associated with the `DescribeSObjectResult`, the `picklistValues` field contains an array of items (`PicklistEntry` objects). Each `PicklistEntry` defines the item's label, value, and whether it is the default item in the picklist (a picklist has no more than one default value).

Enumerated fields support localization of the labels to the language of the user. For example, for the `Industry` field on an `Account`, the value "Agriculture" may be translated to various languages. The enumerated field values are fixed and do not change with a user's language. However, each value may have a specified "label" field that provides the localized label for that value. You must always use the value when inserting or updating a field. The `query()` call always returns the value, not the label. The corresponding label for a value in the `describeSObjectResult` should be used when displaying the value to the user in any user interface.

The API supports the retrieval of the certain picklists in the following objects: `CaseStatus`, `ContractStatus`, `LeadStatus`, `OpportunityStage`, `PartnerRole`, `SolutionStatus`, `TaskPriority`, and `TaskStatus`. Each object represents a value in the respective picklist. These picklist entries always specify some other piece of information, such as whether the status is converted, and so on. Your client application can invoke the `query()` call on any of these objects (such as `CaseStatus`) to retrieve the set of values in the picklist, and then use that information while processing other objects (such as `Case` objects) to find more information about those objects (such as a given case). These objects are read-only via the API. To modify items in picklists, you must use the Salesforce user interface.

Reference Field Type

A reference field contains an `ID` value that points to a unique record (usually the parent record) on another object. This is analogous to the concept of a foreign key in relational databases. The name of a reference field ends, by convention, with the letters `ID` (such as `CaseId` or `OpportunityId`). For example, in the `OpportunityCompetitor` object, the `OpportunityId` field is a reference field that points to the `Opportunity` object. It contains an ID value that uniquely identifies an `Opportunity` record.

In some cases, an object can refer to another object of its same type. For example, an `Account` can have a parent link that points to another `Account`.

The `Event` and `Task` objects both have `WhoId` and `WhatId` cross-reference ID fields. Each of these cross-reference fields can point to one of several other objects. The `WhoId` field can point to a `Contact` or `Lead`, and the `WhatId` field can point to an `Account`, `Opportunity`, `Campaign`, or `Case`. In addition, if the `WhoId` field refers to a `Lead`, then the `WhatId` field must be empty.

You can describe and query each cross-referenced object. When you query a cross-reference ID field, it returns an object ID of the appropriate type. You can then query that ID to get additional information about the object, using the ID in the `id` field for that query.

The cross-reference ID field value is either:

- a valid record in your organization, or
- an empty value, which indicates an empty reference

The cross-reference ID field value, if non-`null`, is guaranteed to be an object in your organization. However, it is not guaranteed that you can query that object. Users with the "View All Data" permission can always query that object. Other users may be restricted from viewing or editing the referenced object.

When specifying a value for a cross-reference ID field in a `create()` or `update()` call, the value must be a valid value of type `ID`, and the user must have appropriate access to that object. The exact requirements vary from field to field.

Textarea Field Type

Textarea fields contain text that can be longer than 4000 bytes. Unlike string fields, textarea fields cannot be specified in the `WHERE` clause of a `queryString` of a `query()` call. To filter records on this field, you must do so while processing records in the `QueryResult`. For fields with this restriction, its `filterable` field in the `Field` type (described in the `fields` property of the `DescribeSObjectResult`) is `false`.

URL Field Type

URL fields contain URLs. Client applications are responsible for specifying valid and properly formatted URLs in `create()` and `update()` calls.

Compound Fields

Compound fields group together multiple elements of primitive data types, such as numbers or strings, to represent complex data types, such as a location or an address. Compound fields are an abstraction that can simplify application code that handles the values, leading to more concise, understandable code.

Address compound fields are available in the SOAP and REST APIs in API version 30.0 and later. Geolocation fields are available in the SOAP and REST APIs in API version 26.0 and later, with some limitations on SOAP for API versions below 30.0.

Compound fields are accessible as a single, structured field, or as individual component fields. The values contained within the compound field and the values in individual fields both map to the same underlying data stored in Salesforce; they always have identical values. Code that references individual component fields is unaffected by the new compound fields.

Compound fields are read-only. Changes are performed by writing to the individual component fields. This maintains a single, consistent method for performing updates, and avoids the possibility of conflicts. For example, if both the `BillingAddress` compound field and `BillingCity` individual component field were updated in the same API call, it would be unclear which value should be saved.

Compound fields are available only through the SOAP and REST APIs. Compound fields are described in both the Enterprise and Partner WSDLs. Update your WSDL to at least API 30.0 to access the new compound data types.

Address Compound Fields

Standard addresses—addresses built into standard objects in Salesforce—are accessible in the SOAP and REST APIs as an `Address`, a structured compound data type, as well as individual address elements.

Using API 30.0 and later, standard addresses are available in the SOAP and REST APIs as a compound field of type `Address`, a structured data type that combines the following fields.

- Accuracy
- City
- Country
- CountryCode
- Latitude
- Longitude
- PostalCode
- State
- StateCode
- Street

 **Note:** `StateCode` and `CountryCode` are always available on compound address fields, whether or not state and country picklists are enabled in your organization.

The `Address` type extends the `Location` type, the data type used for compound geolocation fields.

Address fields are provided on many standard objects, such as Account, Contact, Quote, and User. Some objects provide fields for multiple addresses. For example, Account provides for four different addresses. In this case, address field names are prefixed with the type of address, for example, BillingCity, BillingState, and so on.



Note: Standard address compound fields are read-only, and are only accessible using the SOAP and REST APIs. See [Compound Field Considerations and Limitations](#) on page 42 for additional details of the restrictions this imposes.

When an address is geocoded, its latitude and longitude fields are populated with coordinates. A related geolocation field is also populated. Typically, geocoding service providers geocode addresses, and rate the accuracy of the geocodes.

The accuracy subfield `GeocodeAccuracy` stores the accuracy data for a geocoded location. External geolocation apps can get the accuracy level of a geocoded address via the API. When you retrieve an address via the API, any accuracy data is included. You can also retrieve the accuracy information by itself, if needed.

Like its parent, the compound `Address` field, the `GeocodeAccuracy` field is only available for standard address fields on standard objects.

Retrieving Compound Address Fields

Using compound fields can simplify code that works with addresses, especially for SOQL queries. SOQL `SELECT` clauses can reference addresses directly, instead of all of the individual component fields.

```
SELECT Name, BillingAddress
FROM Account
```

To write code that's compatible with API versions before 30.0, as well as API 30.0 and above, use the individual fields:

```
SELECT Name, BillingStreet, BillingCity, BillingState, BillingPostalCode,
      BillingCountry, BillingLatitude, BillingLongitude
FROM Account
```

Compound address field values are returned as a structured data type, `Address`. Code that works with compound address fields needs to reference the individual components of the returned value. See the code sample below.



Example: Retrieve a Standard Address Compound Field with the SOAP API

The following Java method uses the Salesforce SOAP API to retrieve and display the Mailing Address for a list of contacts.

```
// Modified version of code in the SOAP API QuickStart
private void querySample() {
    String soqlQuery = "SELECT FirstName, LastName, MailingAddress FROM Contact";
    try {
        QueryResult qr = connection.query(soqlQuery);
        boolean done = false;

        if (qr.getSize() > 0) {
            System.out.println("\nLogged-in user can see "
                + qr.getRecords().length + " contact records.");

            while (!done) {
                System.out.println("");
                SObject[] records = qr.getRecords();
                for (int i = 0; i < records.length; ++i) {
                    Contact con = (Contact) records[i];
                    String fName = con.getFirstName();
                    String lName = con.getLastName();
```

```

        // Access the compound address field MailingAddress
        Address addr = (Address) con.getMailingAddress();
        String streetAddr = "";
        if (null != addr) streetAddr = addr.getStreet();

        if ( fName == null) {
            System.out.println("Contact " + (i + 1) + ": " + lName +
                " -- " + streetAddr);
        } else {
            System.out.println("Contact " + (i + 1) + ": " + fName +
                " " + lName +
                " -- " + streetAddr);
        }
    }

    if (qr.isDone()) {
        done = true;
    } else {
        qr = connection.queryMore(qr.getQueryLocator());
    }
}
} else {
    System.out.println("No records found.");
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

```

Using Compound Address Fields as Locations

Compound address fields include latitude and longitude fields. Address fields can be used as locations in SOQL WHERE and ORDER BY clauses. For example, here's a SOQL query that uses the GEOLOCATION function to retrieve the 10 accounts closest to San Francisco.

```
SELECT Id, Name, BillingAddress
FROM Account
WHERE DISTANCE(BillingAddress, GEOLOCATION(37.775,-122.418), 'mi') < 20
ORDER BY DISTANCE(BillingAddress, GEOLOCATION(37.775,-122.418), 'mi')
LIMIT 10
```



Note: In Developer, Professional, Enterprise, Unlimited, and Performance editions, Salesforce can automatically add or update geolocation fields for Account, Contact, Lead, and WorkOrder records. To use this feature, your administrator must enable the geo clean rule for each object. For all other objects and editions, set values for latitude and longitude by using SOQL, Workbench, SOAP or REST API, or a geocoding service. You can then use address fields as locatable values. To find geocoding services, search the AppExchange.

Geolocation Compound Field

Geolocation fields are accessible in the SOAP and REST APIs as a `Location`—a structured compound data type—or as individual latitude and longitude elements.

In API versions 26.0 and later, geolocation fields are available in the SOAP and REST APIs as a compound field of type `Location`. This structured data type contains the following fields.

- `latitude`
- `longitude`

 **Note:** SOAP calls that use API versions earlier than 30.0 return geolocation compound values as strings. See "Returned Geolocation Data Types" later in this topic.

Geolocation fields are provided on many standard objects, such as Account, Contact, Quote, and User, as part of their address field or fields. Geolocation fields can also be added as custom fields to standard or custom objects.

 **Note:**

- A geolocation compound field is read-only, although its `latitude` and `longitude` subfields are editable. You can only access compound fields using the SOAP or REST API. For more information about working with compound fields and their subfields, see [Compound Field Considerations and Limitations](#) on page 42.
- Although geolocation fields appear as a single field in the user interface, custom geolocation fields count as *three* custom fields towards your organization's limits: one for latitude, one for longitude, and one for internal use.

Retrieving Compound Geolocation Fields

Using compound fields can simplify code that works with geolocations, especially for SOQL queries. SOQL `SELECT` clauses can reference geolocations directly, instead of the individual component fields.

```
SELECT location__c
FROM Warehouse__c
```

To write code that's compatible with API versions earlier than 26.0 and with API versions 26.0 and later, use the individual latitude and longitude fields.

```
SELECT location__latitude__s, location__longitude__s
FROM Warehouse__c
```

Returned Geolocation Data Types

A compound geolocation field value is returned as the structured data type `Location`. Code that works with compound geolocation fields must reference the individual components of the returned value. See the sample code in [Address Compound Fields](#) on page 39.

In API versions earlier than 30.0, SOAP calls return compound geolocation field values as strings, instead of as a structured data type, for backward compatibility. If you plan to display your latitude and longitude values or pass them to a service that expects strings, use the values that are returned. If you plan to use the values in mathematical calculations or pass them to a map service that expects numbers, cast the results to numbers.

The string value format is:

```
API location: [latitudeValue longitudeValue]
```

An example of a regular expression to parse out the latitude and longitude values is:

```
API location: \[([-+]?\d{1,2}([.]\d+)?) ([+-]?\d{1,3}([.]\d+)?)\]
```

The first capture is the latitude, and the third is the longitude.

Compound Field Considerations and Limitations

Address and geolocation compound fields are convenient and result in more concise, clear code. Here are some things to consider when using them in your apps.

Both address and geolocation compound fields have the following limitations.

- Compound fields are read-only. To update field values, modify the individual field components.
- Compound fields are accessible only through the SOAP and REST APIs. The compound versions of fields aren't accessible anywhere in the Salesforce user interface.
- Although compound fields can be queried with the `Location` and `Address` Apex classes, they're editable only as components of the actual field. Read and set geolocation field components by appending “`_latitude__s`” or “`_longitude__s`” to the field name, instead of the usual “`_c`.” For example:

```
Double theLatitude = myObject__c.aLocation__latitude__s;  
myObject__c.aLocation__longitude__s = theLongitude;
```

You can't access or set the compound value.

- You can't use compound fields in Visualforce—for example, in an `<apex:outputField>`. To access or update field values, use the individual field components.
- If you select compound fields for export in the Data Loader, they cause error messages. To export values, use individual field components.
- Custom geolocation and location fields on standard addresses aren't supported with email templates.
- You can't use compound fields in lookup filters, except to filter distances that are within or not within given ranges. You can use distance lookup filters only in the Metadata API.
- The only formula functions that you can use with compound fields are `ISBLANK`, `ISCHANGED`, and `ISNULL`. You can't use `BLANKVALUE`, `CASE`, `NULLVALUE`, `PRIORVALUE`, or the equality and comparison operators with compound fields. The equality and comparison operators include `=` and `==` (equal), `<>` and `!=` (not equal), `<` (less than), `>` (greater than), `<=` (less than or equal), `>=` (greater than or equal), `&&` (AND), and `||` (OR).

Address compound fields have the following limitations.

- Compound address fields are available only for address fields that exist as part of the standard objects included in Salesforce. You can't create custom compound address fields.
- In Developer, Professional, Enterprise, Unlimited, and Performance editions, Salesforce can automatically add or update geolocation fields for Account, Contact, Lead, and WorkOrder records. To use this feature, your administrator must enable the geo clean rule for each object. For all other objects and editions, set values for latitude and longitude by using SOQL, Workbench, SOAP or REST API, or a geocoding service. You can then use address fields as locatable values. To find geocoding services, search the AppExchange.
- The accuracy subfield of address fields is populated only when an address is geocoded. Typically, geocoding service providers provide accuracy data for an address's latitude and longitude coordinates.
- Address fields can't be used in `WHERE` statements in SOQL. Address fields aren't filterable, but the `isFilterable()` method of the `DescribeFieldResult` Apex class erroneously returns `true` for address fields.

Geolocation compound fields have the following limitations.

- Geolocation fields aren't supported in custom settings.
- Geolocation fields aren't available in dashboards or Schema Builder.
- Geolocation fields are available in Visual Workflow and in formula-based workflow and approvals, but they can't be used in filter-based workflow updates and approvals.
- `DISTANCE` formulas are supported in:
 - Entry criteria for workflow rules and approval processes

- Field update actions in workflow rules and approval processes
- Custom validation rules
- Lookup filters (in the Metadata API only)
- Geolocation fields and latitude and longitude on standard addresses aren't supported in Salesforce to Salesforce.
- In Developer, Professional, Enterprise, Unlimited, and Performance editions, Salesforce can automatically add or update geolocation fields for Account, Contact, Lead, and WorkOrder records. To use this feature, your administrator must enable the geo clean rule for each object. For all other objects and editions, set values for latitude and longitude by using SOQL, Workbench, SOAP or REST API, or a geocoding service. You can then use address fields as locatable values. To find geocoding services, search the AppExchange.
- Geolocation fields are supported in SOQL with the following limitations.
 - DISTANCE and GEOLOCATION are supported in WHERE and ORDER BY clauses in SOQL, but not in GROUP BY. DISTANCE is supported in SELECT clauses.
 - DISTANCE supports only the logical operators > and <, returning values within (<) or beyond (>) a specified radius.
 - When using the GEOLOCATION function in SOQL queries, the geolocation field must precede the latitude and longitude coordinates. For example, DISTANCE (warehouse_location_c, GEOLOCATION (37.775, -122.418), 'km') works but DISTANCE (GEOLOCATION (37.775, -122.418), warehouse_location_c, 'km') doesn't work.
 - Apex bind variables aren't supported for the units parameter in DISTANCE or GEOLOCATION functions. This query doesn't work.

```
String units = 'mi';
List<Account> accountList =
    [SELECT ID, Name, BillingLatitude, BillingLongitude
     FROM Account
     WHERE DISTANCE (My_Location_Field_c, GEOLOCATION(10,10), :units) < 10];
```

For more information and examples, see the [Force.com SOQL and SOSL Reference](#).

API Data Types and Salesforce Field Types

Generally, API data types and field types in the user interface have the same names. For example, a date field is represented by a date data type in the API. However, some field types are represented differently depending on whether you are inspecting an object via the API or the user interface. The following table contains the mapping for field types and data types that are different:

API Data Type	Corresponding Field Types in the User Interface
ID	Lookup relationship, master-detail relationship
string	Auto number, email, phone, picklist, multi-select picklist, text, text area, long text area, rich text area, data category group reference and URL. Different maximum lengths are specified in the WSDL for text, text area, and long text area.
boolean	Checkbox
double	Currency, formula, number, percent, and roll-up summary
Varies by type	When formula fields are created in the user interface, a type must be specified. This type corresponds to the API data type of the same name: currency, date, date/time, number, percent, or text.

All other fields that you can create in the user interface fall into one of the following categories:

- The field is not available in both the user interface and the API. For example, the BusinessHours object has fields of API data type time, but you cannot create a custom field of this type.
- Field types are the same as their corresponding API data type. For example, if you create a date field in the user interface, that field is the date data type in the API.

For more information about API data types, see [Primitive Data Types](#) and [Field Types](#).

Core Data Types Used in API Calls

Many calls in the API use the following data types:

- [sObject](#)
- ID (String). See [ID Field Type](#).

The API also uses several error handling objects. If an error occurs during a SOAP request, the API returns a SOAP fault message. The message contains different content, depending on the type of error:

- If an error affects the entire request, an [API Fault Element](#), is returned, containing an [ExceptionCode](#) and the associated error message text.
- If the error affects some records and not others, an [Error](#) is returned, containing a [StatusCode](#). These errors typically occur during bulk operations, such as creating, updating, or deleting multiple records with a single call.

sObject

An sObject represents an object, such as an [Account](#) or [Campaign](#). For a list of standard objects, see [Standard Objects](#).

An sObject has the following properties:

Name	Type	Description
fieldsToNull	string[]	<p>Array of one or more field names whose value you want to explicitly set to <code>null</code>.</p> <p>When used with <code>update()</code> or <code>upsert()</code>, you can specify only those fields that you can update and that have the <code>nillable</code> property. When used with <code>create()</code>, you can specify only those fields that you can create and that have the <code>nillable</code> or the <code>default</code> on <code>create</code> property.</p> <p>For example, if specifying an ID field or required field results in a runtime error, you can specify that field name in <code>fieldsToNull</code>. Similarly, if a picklist field has a default value and you want to set the value to <code>null</code> instead, specify the field in <code>fieldsToNull</code>.</p>
ID	ID	Unique ID for this individual object. For the <code>create()</code> call, this value is null. For all other API calls, this value must be specified.

API Fault Element

An `ApiFault` element contains information about a fault that occurs when processing a service request. The `ApiFault` element has the following properties.

Name	Type	Description
exceptionCode	ExceptionCode	A code that characterizes the exception. The full list of exception codes is available in the WSDL file for your org. See Generating the WSDL File for Your Organization .
exceptionMessage	string	Exception message text.
extendedErrorDetails	ExtendedErrorDetails	More details about the exception, including an extended error code and extra error properties, when available. Reserved for future use.

The following table lists the API fault elements that represent all the API faults that can occur.

Fault	Description
ApiQueryFault	The row and column numbers where the problem occurred.
LoginFault	An error occurred during the <code>login()</code> call.
InvalidSObjectFault	An invalid <code>sObject</code> in a <code>describeSObject()</code> , <code>describeSObjects()</code> , <code>describeLayout()</code> , <code>describeDataCategoryGroups()</code> , <code>describeDataCategoryGroupStructures()</code> , <code>create()</code> , <code>update()</code> , <code>retrieve()</code> , or <code>query()</code> call.
InvalidFieldFault	An invalid field in a <code>retrieve()</code> or <code>query()</code> call.
InvalidOrNullForRestrictedPicklist	An invalid <code>appMenuItemType</code> in a <code>describeAppMenu()</code> call.
MalformedQueryFault	A problem in the <code>queryString</code> passed in a <code>query()</code> call.
InvalidQueryLocatorFault	A problem in the <code>queryLocator</code> passed in a <code>queryMore()</code> call.
MalformedSearchFault	A problem in the <code>search</code> passed in a <code>search()</code> call.
InvalidIdFault	A specified ID was invalid in a <code>setPassword()</code> or <code>resetPassword()</code> call.
UnexpectedErrorFault	An unexpected error occurred. The error is not associated with any other API fault.

ExceptionCode

The following list of `ExceptionCode` values is defined in your WSDL file. Some codes don't appear in your WSDL, depending on the features you have enabled.

API_CURRENTLY_DISABLED

Because of a system problem, API functionality is temporarily unavailable.

API_DISABLED_FOR_ORG

API access has not been enabled for the org. Contact Salesforce to enable API access.

CANT_ADD_STANDARD_PORTAL_USER_TO_TERRITORY

A user with a standard portal license can't be added to a territory.

CIRCULAR_OBJECT_GRAPH

The request failed because it contained a circular object reference.

CLIENT_NOT_ACCESSIBLE_FOR_USER

The current user does not have permission to access the specified client.

CLIENT_REQUIRE_UPDATE_FOR_USER

The current user is required to use a newer version of the specified client and doesn't have access until the client is updated.

DELETE_REQUIRED_ON CASCADE

The delete operation triggers a cascade delete on a record, but the logged-in user does not have delete permission on that related object.

DUPLICATE_COMM_NICKNAME

You can't create a user with the same nickname as another user.

DUPLICATE_VALUE

You can't supply a duplicate value for a field that must be unique. For example, you can't submit two copies of the same session ID in a `invalidateSessions()` call.

EMAIL_BATCH_SIZE_LIMIT_EXCEEDED

A method tried to process more email records than the maximum batch size.

EMAIL_TO_CASE_INVALID_ROUTING

An email to case record has been submitted for processing but the feature is not enabled.

EMAIL_TO_CASE_LIMIT_EXCEEDED

The daily converted email limit for the Email-to-Case feature has been exceeded.

EMAIL_TO_CASE_NOT_ENABLED

The Email-to-Case feature has not been enabled.

EXCEEDED_ID_LIMIT

Too many IDs were specified in a call. For example, more than 2000 IDs were requested in a `retrieve()` call, or more than 200 session IDs were specified in a `logout()` call.

EXCEEDED_LEAD_CONVERT_LIMIT

Too many IDs were sent to a `convertLead()` call.

EXCEEDED_MAX_SIZE_REQUEST

The size of the message sent to the API exceeded 50 MB.

EXCEEDED_MAX_TYPES_LIMIT

The number of object types to describe is too large.

EXCEEDED_QUOTA

The size limit for org data storage was exceeded during a `create()` call.

FUNCTIONALITY_NOT_ENABLED

Functionality has been temporarily disabled. Other calls continue to work.

INACTIVE_OWNER_OR_USER

The user or record owner is not active.

INACTIVE_PORTAL

The referenced portal is inactive.

INSUFFICIENT_ACCESS

The user does not have sufficient access to perform the operation.

INVALID_ASSIGNMENT_RULE

An invalid `AssignmentRuleHeader` value was specified.

INVALID_BATCH_SIZE

The query options have an invalid batch size value.

INVALID_CLIENT

The client is invalid.

INVALID_CROSS_REFERENCE_KEY

An invalid foreign key can't be set on a field. For example, an object share, such as AccountShare, can't be deleted because the share row is a result of a sharing rule.

INVALID_FIELD

The specified field name is invalid.

INVALID_FILTER_LANGUAGE

The specified language can't be used as a filter.

INVALID_FILTER_VALUE

A SOQL query with `LIKE` specified an invalid character, for example, an incorrectly placed asterisk (*). Correct the query and resubmit.

INVALID_ID_FIELD

The specified ID is correctly formatted but isn't valid. For example, the ID is of the wrong type, or the object it identifies no longer exists.

INVALID_GOOGLE_DOCS_URL

An invalid Salesforce record URL was used when trying to associate a Google Doc to that record. Correct the URL before trying the operation again.

INVALID_LOCATOR

The locator is invalid.

INVALID_LOGIN

The `login()` credentials are not valid, or the maximum number of logins have been exceeded. Contact your administrator for more information.

INVALID_NEW_PASSWORD

The new password does not conform with the password policies of the org.

INVALID_OPERATION

The client application tried to modify a record that is locked by an approval process.

INVALID_OPERATION_WITH_EXPIRED_PASSWORD

Due to password expiration, a valid password must be set using `setPassword()` before the call can be invoked.

INVALID_QUERY_FILTER_OPERATOR

An invalid operator was used in the `query()` filter clause, at least for that field.

INVALID_QUERY_LOCATOR

An invalid `queryLocator` parameter was specified in a `queryMore()` call. It's also possible that you've exceed the maximum number of calls, which is 10 per user for the API, and 5 for Apex and Visualforce.

INVALID_QUERY_SCOPE

The specified search scope is invalid.

INVALID_REPLICATION_DATE

The date for replication is out of the allowed range, such as before the org was created.

INVALID_SETUP_OWNER

The setup owner must be an Organization, Profile, or User.

INVALID_SEARCH

The `search()` call has invalid syntax or grammar. For more information, see the [Salesforce SOQL and SOSL Reference Guide](#).

INVALID_SEARCH_SCOPE

The specified search scope is invalid.

INVALID_SESSION_ID

The specified [sessionId](#) is malformed (incorrect length or format) or has expired. Log in again to start a new session.

INVALID_SOAP_HEADER

There is an error in the SOAP header. If you are migrating from an earlier version of the API, be advised that the SaveOptions header can't be used with API version 6.0 or later. Use [AssignmentRuleHeader](#) instead.

INVALID_SSO_GATEWAY_URL

The URL provided to configure the Single Sign-On gateway was not a valid URL.

INVALID_TYPE

The specified [sObject](#) type is invalid.

INVALID_TYPE_FOR_OPERATION

The specified [sObject](#) type is invalid for the specified operation.

LIMIT_EXCEEDED

An array is too long. For example, there are too many BCC addresses, targets, or email messages.

LOGIN_DURING_RESTRICTED_DOMAIN

The user is not allowed to log in from this IP address.

LOGIN_DURING_RESTRICTED_TIME

The user is not allowed to log in during this time period.

MALFORMED_ID

An invalid ID string was specified. For information about IDs, see [ID Field Type](#).

MALFORMED_QUERY

An invalid query string was specified. For example, the query string was longer than 20,000 characters.

MALFORMED_SEARCH

An invalid search string was specified. For example, the search string was longer than 20,000 characters.

MISSING_ARGUMENT

A required argument is missing.

MIXED_DML_OPERATION

There are limits on what kinds of DML operations can be performed in the same transaction. For more information, see the [Force.com Apex Code Developer's Guide](#).

NOT_MODIFIED

The describe call response has not changed since the specified date.

NO_SOFTPHONE_LAYOUT

If an org has the CTI feature enabled, but no softphone layout has been defined, this exception is returned if a describe call is issued. This exception is most often caused because no call center has been defined. A default softphone layout is created during call center definition.

If an org doesn't have the CTI feature enabled, [FUNCTIONALITY_NOT_ENABLED](#) is returned instead.

NUMBER_OUTSIDE_VALID_RANGE

The number specified is outside the valid range for the field.

OPERATION_TOO_LARGE

The query has returned too many results. If certain queries are run by a user without the "View All Data" permission and many records are returned, the queries require sharing rule checking. For example, consider queries that are run on objects, such as [Task](#), that use

a polymorphic foreign key. These queries return this exception because the operation requires too many resources. To correct, add filters to the query to narrow the scope, or use filters such as date ranges to break the query up into a series of smaller queries.

ORG_LOCKED

The org has been locked. Contact Salesforce to unlock the org.

ORG_NOT OWNED_BY_INSTANCE

The user tried to log in to the wrong server instance. Choose another server instance or log in at <https://login.salesforce.com>. You can use `http` instead of `https`.

PASSWORD_LOCKOUT

The user has exceeded the allowed number of login attempts. The user must contact an administrator to regain login access.

PORTAL_NO_ACCESS

Access to the specified portal is not available.

QUERY_TIMEOUT

The query has timed out. For more information, see the [Salesforce SOQL and SOSL Reference Guide](#).

QUERY_TOO_COMPLEX

SOQL query is either selecting too many fields or there are too many filter conditions. Try reducing the number of formula fields referenced in the query.

REQUEST_LIMIT_EXCEEDED

Exceeded either the concurrent request limit or the request rate limit for your org. For details on API request limits, see [API Usage Metering](#).

REQUEST_RUNNING_TOO_LONG

A request has taken too long to be processed.

SERVER_UNAVAILABLE

A server that is necessary for this call is unavailable. Other types of requests could still work.

SSO_SERVICE_DOWN

The service was unavailable, and an authentication call to the org's specified Single Sign-On server failed.

TOO_MANY_APEX_REQUESTS

Too many Apex requests have been issued. If this exception persists, contact Salesforce Customer Support.

TRIAL_EXPIRED

The trial period for the org has expired. A representative from the company must contact Salesforce to re-enable the org.

UNSUPPORTED_API_VERSION

A method call was made that doesn't exist in the accessed API version, for example, trying to use `upsert()` (new in 8.0) against version 5.0.

UNSUPPORTED_CLIENT

This version of the client is no longer supported.

Error

An `Error` contains information about an error that occurred during a `create()`, `merge()`, `process()`, `update()`, `upsert()`, `delete()`, or `undelete()` call. For more information, see [Error Handling](#). An `Error` has the following properties:

Name	Type	Description
<code>statusCode</code>	<code>StatusCode</code>	A code that characterizes the error. The full list of status codes is available in the WSDL file for your org (see Generating the WSDL File for Your Organization).

Name	Type	Description
message	string	Error message text.
fields	string[]	Array of one or more field names. Identifies which fields in the object, if any, affected the error condition.
extendedErrorDetails	ExtendedErrorDetails	More details about the error, including an extended error code and extra error properties, when available. Reserved for future use.



Note: If your org has active duplicate rules and a duplicate is detected, the SaveResult includes an error with a data type of [DuplicateError](#).

StatusCode

The following table lists API status codes that are returned with an error. Some codes don't appear in your WSDL, depending on the features you have enabled.

ALL_OR_NONE_OPERATION_ROLLED_BACK

The bulk operation was rolled back because one of the records wasn't processed successfully. See [AllOrNoneHeader](#).

ALREADY_IN_PROCESS

You can't submit a record that is already in an approval process. Wait for the previous approval process to complete before resubmitting a request with this record.

ASSIGNEE_TYPE_REQUIRED

Designate an assignee for the approval request (ProcessInstanceStep or ProcessInstanceWorkitem).

BAD_CUSTOM_ENTITY_PARENT_DOMAIN

The changes you are trying to make can't be completed because changes to the associated master-detail relationship can't be made.

BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED

Your client application blind carbon-copied an email address even though the org's Compliance BCC Email option is enabled. This option specifies a particular email address that automatically receives a copy of all outgoing email. When this option is enabled, you can't BCC any other email address. To disable the option, log in to the user interface and from Setup, enter *Compliance BCC Email* in the Quick Find box, then select **Compliance BCC Email**.

BCC_SELF_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED

Your client application blind carbon-copied the logged-in user's email address even though the org's BCC COMPLIANCE option is set to true. This option specifies a particular email address that automatically receives a copy of all outgoing email. When this option is enabled, you can't BCC any other email address. To disable the option, log in to the user interface and from Setup, enter *Compliance BCC Email* in the Quick Find box, then select **Compliance BCC Email**.

CANNOT.Cascade_PRODUCT_ACTIVE

An update to a product caused by a cascade can't be done because the associated product is active.

CANNOT_CHANGE_FIELD_TYPE_OF_APEX_REFERENCED_FIELD

You can't change the type of a field that is referenced in an Apex script.

CANNOT_CREATE_ANOTHER_MANAGED_PACKAGE

You can create only one managed package in an org.

CANNOT_DEACTIVATE_DIVISION

You can't deactivate Divisions if an assignment rule references divisions or if the `DefaultDivision` field on a user record isn't set to null.

CANNOT_DELETE_LAST_DATED_CONVERSION_RATE

If dated conversions are enabled, you must have at least one DatedConversionRate record.

CANNOT_DELETE_MANAGED_OBJECT

You can't modify components that are included in a managed package.

CANNOT_DISABLE_LAST_ADMIN

You must have at least one active administrator user.

CANNOT_ENABLE_IP_RESTRICT_REQUESTS

If you exceed the limit of five IP ranges specified in a profile, you can't enable restriction of login by IP addresses. Reduce the number of specified ranges in the profile and try the request again.

CANNOT_INSERT_UPDATE_ACTIVATE_ENTITY

You do not have permission to create, update, or activate the specified record.

CANNOT_MODIFY_MANAGED_OBJECT

You can't modify components that are included in a managed package.

CANNOT_RENAME_APEX_REFERENCED_FIELD

You can't rename a field that is referenced in an Apex script.

CANNOT_RENAME_APEX_REFERENCED_OBJECT

You can't rename an object that is referenced in an Apex script.

CANNOT_REPARENT_RECORD

You can't define a new parent record for the specified record.

CANNOT_RESOLVE_NAME

A sendEmail() call could not resolve an object name.

CANNOT_UPDATE_CONVERTED_LEAD

A converted lead could not be updated.

CANT_DISABLE_CORP_CURRENCY

You can't disable the corporate currency for an org. To disable a currency that is set as the corporate currency, first use the user interface to change the corporate currency to a different currency. Then disable the original currency.

CANT_UNSET_CORP_CURRENCY

You can't change the corporate currency for an org from the API. Use the user interface to change the corporate currency.

CHILD_SHARE_FAILS_PARENT

If you don't have appropriate permissions on a parent record, you can't change the owner of or define sharing rules for a child record. For example, you can't change the owner of a contact record if you can't edit its parent account record.

CIRCULAR_DEPENDENCY

You can't create a circular dependency between metadata objects in your org. For example, public group A can't include public group B, if public group B already includes public group A.

COMMUNITY_NOT_ACCESSIBLE

You do not have permission to access the community that this entity belongs to. You must be given permission to access the community before you can access this entity.

CUSTOM_CLOB_FIELD_LIMIT_EXCEEDED

You can't exceed the maximum size for a CLOB field.

CUSTOM_ENTITY_OR_FIELD_LIMIT

You have reached the maximum number of custom objects or custom fields for your org.

CUSTOM_FIELD_INDEX_LIMIT_EXCEEDED

You have reached the maximum number of indexes on a field for your org.

CUSTOM_INDEX_EXISTS

You can create only one custom index per field.

CUSTOM_LINK_LIMIT_EXCEEDED

You have reached the maximum number of custom links for your org.

CUSTOM_METADATA_LIMIT_EXCEEDED

Your org has reached its custom metadata maximum limit.

CUSTOM_SETTINGS_LIMIT_EXCEEDED

Your org has reached its custom settings maximum limit.

CUSTOM_TAB_LIMIT_EXCEEDED

You have reached the maximum number of custom tabs for your org.

DELETE_FAILED

You can't delete a record because it is in use by another object.

DEPENDENCY_EXISTS

You can't perform the requested operation because of an existing dependency on the specified object or field.

DUPLICATE_CASE SOLUTION

You can't create a relationship between the specified case and solution because it already exists.

DUPLICATE_CUSTOM_ENTITY_DEFINITION

Custom object or custom field IDs must be unique.

DUPLICATE_CUSTOM_TAB_MOTIF

You can't create a custom object or custom field with a duplicate master name.

DUPLICATE_DEVELOPER_NAME

You can't create a custom object or custom field with a duplicate developer name.

DUPLICATES_DETECTED

Duplicate records have been detected. Used for an Error object with a data type of [DuplicateError](#).

DUPLICATE_EXTERNAL_ID

A user-specified external ID matches more than one record during an upsert.

DUPLICATE_MASTER_LABEL

You can't create a custom object or custom field with a duplicate master name.

DUPLICATE_SENDER_DISPLAY_NAME

A `sendEmail()` call could not choose between `OrgWideEmailAddress.displayName` or `senderDisplayName`. Define only one of the two fields.

DUPLICATE_USERNAME

A create, update, or upsert failed because of a duplicate user name.

DUPLICATE_VALUE

You can't supply a duplicate value for a field that must be unique. For example, you can't submit two copies of the same session ID in a `invalidateSessions()` call.

EMAIL_ADDRESS_BOUNCED

Emails to one or more recipients have bounced. Check the email addresses to make sure that they are valid.

EMAIL_NOT_PROCESSED_DUE_TO_PRIOR_ERROR

Because of an error earlier in the call, the current email was not processed.

EMAIL_OPTED_OUT

A single email message was sent with the `REJECT` setting in the `optOutPolicy` field to recipients that have opted out from receiving email. To avoid this error, set the `optOutPolicy` field to another value.

EMAIL_TEMPLATE_FORMULA_ERROR

The email template is invalid and can't be rendered. Check the template for incorrectly specified merge fields.

EMAIL_TEMPLATE_MERGEFIELD_ACCESS_ERROR

You don't have access to one or more merge fields in this template. To request access, contact your Salesforce administrator.

EMAIL_TEMPLATE_MERGEFIELD_ERROR

One or more merge fields don't exist. Check the spelling of field names.

EMAIL_TEMPLATE_MERGEFIELD_VALUE_ERROR

One or more merge fields have no value. To provide values, update the records before sending the email.

EMAIL_TEMPLATE_PROCESSING_ERROR

The merge fields in this email template can't be processed. Ensure that your template body is valid.

EMPTY_SCONTROL_FILE_NAME

The Scontrol file name was empty, but the binary was not empty.

ENTITY_FAILED_IFLASTMODIFIED_ON_UPDATE

If the value in a record's `LastModifiedDate` field is later than the current date, you can't update the record.

ENTITY_IS_ARCHIVED

If a record has been archived, you can't access it.

ENTITY_IS_DELETED

You can't reference an object that has been deleted. This status code occurs only in API version 10.0 and later. Previous releases of the API use `INVALID_ID_FIELD` for this error.

ENTITY_IS_LOCKED

You can't edit a record that is locked by an approval process.

ENVIRONMENT_HUB_MEMBERSHIP_CONFLICT

You can't add an org to more than one Environment Hub.

ERROR_IN_MAILER

An email address is invalid, or another error occurred during an email-related transaction.

FAILED_ACTIVATION

The activation of a Contract failed.

FIELD_CUSTOM_VALIDATION_EXCEPTION

You can't define a custom validation formula that violates a field integrity rule.

FIELD_FILTER_VALIDATION_EXCEPTION

You can't violate field integrity rules.

FILTERED_LOOKUP_LIMIT_EXCEEDED

The creation of the lookup filter failed because it exceeds the maximum number of lookup filters allowed per object.

HTML_FILE_UPLOAD_NOT_ALLOWED

Your attempt to upload an HTML file failed. HTML attachments and documents, including HTML attachments to a [Solution](#), can't be uploaded if the `Disallow HTML documents and attachments` checkbox is selected on the HTML Documents and Attachments Settings page.

IMAGE_TOO_LARGE

The image exceeds the maximum width, height, and file size.

INACTIVE_OWNER_OR_USER

The owner of the specified item is an inactive user. To reference this item, either reactivate the owner or reassign ownership to another active user.

INSERT_UPDATE_DELETE_NOT_ALLOWED_DURING_MAINTENANCE

Starting with version 32.0, you can't create, update, or delete data while the instance where your org resides is being upgraded to the latest release. Try again after the release has completed. For release schedules, see trust.salesforce.com. Before version 32.0, the code is `INVALID_READ_ONLY_USER_DML`.

INSUFFICIENT_ACCESS_ON_CROSS_REFERENCE_ENTITY

An operation affects an object that is cross-referenced by the specified object, but the logged-in user doesn't have sufficient permissions on the cross-referenced object. For example, a logged-in user attempts to modify an account record, and the update creates a ProcessInstanceWorkitem. If the user doesn't have permission to approve, reject, or reassign the ProcessInstanceWorkitem, this exception occurs.

INSUFFICIENT_ACCESS_OR_READONLY

You can't perform the specified action because you don't have sufficient permissions.

INVALID_ACCESS_LEVEL

You can't define a new sharing rule that provides less access than the specified org-wide default.

INVALID_ARGUMENT_TYPE

You supplied an argument that is of the wrong type for the operation being attempted.

INVALID_ASSIGNEE_TYPE

You specified an assignee type that is not a valid integer between one and six.

INVALID_ASSIGNMENT_RULE

You specified an assignment rule that is invalid or that isn't defined in the org.

INVALID_BATCH_OPERATION

The specified batch operation is invalid.

INVALID_CONTENT_TYPE

The outgoing email has anEmailFileAttachment with an `invalidContentType` property. See [RFC2045 - Internet Message Format](#).

INVALID_CREDIT_CARD_INFO

The specified credit card information is not valid.

INVALID_CROSS_REFERENCE_KEY

The specified value in a relationship field is not valid, or data is not of the expected type.

INVALID_CROSS_REFERENCE_TYPE_FOR_FIELD

The specified cross-reference type is not valid for the specified field.

INVALID_CURRENCY_CONV_RATE

Specify a positive, non-zero value for the currency conversion rate.

INVALID_CURRENCY_CORP_RATE

You can't modify the corporate currency conversion rate.

INVALID_CURRENCY_ISO

The specified [currency ISO code](#) is not valid.

INVALID_EMAIL_ADDRESS

A specified email address is invalid.

INVALID_EMPTY_KEY_OWNER

You can't set the value for `owner` to null.

INVALID_EVENT_SUBSCRIPTION

Invalid parameters were specified when subscribing to an event.

INVALID_FIELD

You specified an invalid field name when trying to update or upsert a record.

INVALID_FIELD_FOR_INSERT_UPDATE

You can't combine a person account record type change with any other field update.

INVALID_FIELD_WHEN_USING_TEMPLATE

You can't use an email template with an invalid field name.

INVALID_FILTER_ACTION

The specified filter action can't be used with the specified object. For example, an alert is not a valid filter action for a Task.

INVALID_ID_FIELD

The specified ID field (`ID`, `ownerId`), or cross-reference field is invalid.

INVALID_INET_ADDRESS

A specified Inet address is not valid.

INVALID_LINEITEM_CLONE_STATE

You can't clone a Pricebook2 or PricebookEntry record that isn't active.

INVALID_MASTER_OR_TRANSLATED SOLUTION

The solution is invalid. For example, this exception occurs if you try to associate a translated solution with a master solution that's associated with another translated solution.

INVALID_MESSAGE_ID_REFERENCE

The outgoing email's References or In-Reply-To fields are invalid. These fields must contain valid Message-IDs. See [RFC2822 - Internet Message Format](#).

INVALID_OPERATION

There is no applicable approval process for the specified object.

INVALID_OPERATOR

The specified operator is not applicable for the field type when used as a workflow filter.

INVALID_OR_NULL_FOR_RESTRICTED_PICKLIST

You specified an invalid or null value for a restricted picklist.

INVALID_PARTNER_NETWORK_STATUS

The specified partner network status is invalid for the specified template field.

INVALID_PERSON_ACCOUNT_OPERATION

You can't delete a person account.

INVALID_READ_ONLY_USER_DML

Version 31.0 and earlier: You can't create, update, or delete data while the instance where your org resides is being upgraded to the latest release. Try again after the release has completed. For release schedules, see [trust.salesforce.com](#). After version 31.0, the code is `INSERT_UPDATE_DELETE_NOT_ALLOWED_DURING_MAINTENANCE`.

INVALID_SAVE_AS_ACTIVITY_FLAG

Specify `true` or `false` for the `saveAsActivity` flag.

INVALID_SESSION_ID

The specified `sessionId` is malformed (incorrect length or format) or has expired. Log in again to start a new session.

INVALID_STATUS

The specified org status change is not valid.

INVALID_TYPE

The specified type is not valid for the specified object.

INVALID_TYPE_FOR_OPERATION

The specified type is not valid for the specified operation.

INVALID_TYPE_ON_FIELD_IN_RECORD

The specified value is not valid for the specified field's type.

IP_RANGE_LIMIT_EXCEEDED

The specified IP address is outside the IP range specified for the org.

JIGSAW_IMPORT_LIMIT_EXCEEDED

The number of records you attempted to purchase from Data.com exceeds your available record addition limit.

LICENSE_LIMIT_EXCEEDED

You have exceeded the number of licenses assigned to your org.

LIGHT_PORTAL_USER_EXCEPTION

You attempted an action with a high-volume portal user that's not allowed. For example, trying to add the user to a case team.

LIMIT_EXCEEDED

You have exceeded a limit on a field size or value, license, or other component.

LOGIN_CHALLENGE_ISSUED

An email containing a security token was sent to the user's email address because the user logged in from an untrusted IP address.

The user can't log in until the security token is added to the end of the password.

LOGIN_CHALLENGE_PENDING

The user logged in from an untrusted IP address, but a security token has not yet been issued.

LOGIN_MUST_USE_SECURITY_TOKEN

The user must add a security token to the end of the password to log in.

MALFORMED_ID

An ID must be either 15 characters, or 18 characters with a valid case-insensitive extension. There is also an exception code of the same name.

MANAGER_NOT_DEFINED

A manager has not been defined for the specified approval process.

MASSMAIL_RETRY_LIMIT_EXCEEDED

A mass mail retry failed because your org has exceeded its mass mail retry limit.

MASS_MAIL_LIMIT_EXCEEDED

The org has exceeded its daily limit for mass email. Mass email messages can't be sent again until the next day.

MAXIMUM_CCEMAILS_EXCEEDED

You have exceeded the maximum number of specified CC addresses in a workflow email alert.

MAXIMUM_DASHBOARD_COMPONENTS_EXCEEDED

You have exceeded the document size limit for a dashboard.

MAXIMUM_HIERARCHY_LEVELS_REACHED

You have reached the maximum number of levels in a hierarchy.

MAXIMUM_SIZE_OF_ATTACHMENT

You have exceeded the maximum size of an attachment.

MAXIMUM_SIZE_OF_DOCUMENT

You have exceeded the maximum size of a document.

MAX_ACTIONS_PER_RULE_EXCEEDED

You have exceeded the maximum number of actions per rule.

MAX_ACTIVE_RULES_EXCEEDED

You have exceeded the maximum number of active rules.

MAX_APPROVAL_STEPS_EXCEEDED

You have exceeded the maximum number of approval steps for an approval process.

MAX_FORMULAS_PER_RULE_EXCEEDED

You have exceeded the maximum number of formulas per rule.

MAX_RULES_EXCEEDED

You have exceeded the maximum number of rules for an object.

MAX_RULE_ENTRIES_EXCEEDED

You have exceeded the maximum number of entries for a rule.

MAX_TASK_DESCRIPTION_EXCEEDED

The task description is too long.

MAX_TM_RULES_EXCEEDED

You have exceeded the maximum number of rules per Territory.

MAX_TM_RULE_ITEMS_EXCEEDED

You have exceeded the maximum number of rule criteria per rule for a Territory.

MERGE_FAILED

A merge operation failed.

MISSING_ARGUMENT

You did not specify a required argument.

NONUNIQUE_SHIPPING_ADDRESS

You can't insert a reduction order item if the original order shipping address is different from the shipping address of other items in the reduction order.

NO_APPLICABLE_PROCESS

A process () request failed because the record submitted does not satisfy the entry criteria of any active approval processes for which the user has permission.

NO_ATTACHMENT_PERMISSION

Your org does not permit email attachments.

NO_INACTIVE_DIVISION_MEMBERS

You can't add members to an inactive Division.

NO_MASS_MAIL_PERMISSION

You don't have permission to send the email. You must have "Mass Email" to send mass mail or "Send Email" to send individual email.

NUMBER_OUTSIDE_VALID_RANGE

The number specified is outside the valid range of values.

NUM_HISTORY_FIELDS_BY_SOBJECT_EXCEEDED

The number of history fields specified for the sObject exceeds the allowed limit.

OP_WITH_INVALID_USER_TYPE_EXCEPTION

The operation you attempted can't be performed for one or more users. For example, you can't add high-volume portal users to a group.

OPTED_OUT_OF_MASS_MAIL

An email can't be sent because the specified User has opted out of mass mail.

PACKAGE_LICENSE_REQUIRED

The logged-in user can't access an object that is in a licensed package without a license for the package.

PORTAL_USER_ALREADY_EXISTS_FOR_CONTACT

A create [User](#) operation failed because you can't create a second portal user under a Contact.

PRIVATE_CONTACT_ON_ASSET

You can't have a private contact on an asset.

RECORD_IN_USE_BY_WORKFLOW

You can't access a record that's in use by a workflow or approval process.

REQUEST_RUNNING_TOO_LONG

A request that has been running too long is canceled.

REQUIRED_FIELD_MISSING

A call requires a field that was not specified.

SELF_REFERENCE_FROM_TRIGGER

You can't recursively update or delete the same object from an Apex trigger. This error often occurs when:

- You try to update or delete an object from within its before trigger.
- You try to delete an object from within its after trigger.

This error occurs with both direct and indirect operations. The following is an example of an indirect operation:

1. A request is submitted to update Object A.
2. A `before update` trigger on object A creates an object B.
3. Object A is updated.
4. An `after insert` trigger on object B queries object A and updates it. This update is an indirect update of object A because of the before trigger of object A, so an error is generated.

SHARE_NEEDED_FOR_CHILD_OWNER

If a parent record has a child record that needs a sharing rule, you can't delete the sharing rule for the parent record.

SINGLE_EMAIL_LIMIT_EXCEEDED

(API version 18.0 and later) The org has exceeded its daily limit for individual emails. Individual email messages can't be sent again until the next day.

STANDARD_PRICE_NOT_DEFINED

Custom prices can't be defined without corresponding standard prices.

STORAGE_LIMIT_EXCEEDED

You have exceeded your org's storage limit.

STRING_TOO_LONG

The specified string exceeds the maximum allowed length.

TABSET_LIMIT_EXCEEDED

You have exceeded the number of tabs allowed for a tabset.

TEMPLATE_NOT_ACTIVE

The template specified is unavailable. Specify another template or make the template available for use.

TERRITORY_REALIGN_IN_PROGRESS

An operation can't be performed because a territory realignment is in progress.

TEXT_DATA_OUTSIDE_SUPPORTED_CHARSET

The specified text uses a character set that is not supported.

TOO_MANY_APEX_REQUESTS

Too many Apex requests have been sent. This error is transient. Resend your request after a short wait.

TOO_MANY_ENUM_VALUE

A request failed because too many values were passed in for a multi-select picklist. You can select a maximum of 100 values for a multi-select picklist.

TRANSFERQUIRES_READ

You can't assign the record to the specified User because the user does not have read permission.

UNABLE_TO_LOCK_ROW

A deadlock or timeout condition has been detected.

- A deadlock involves at least two transactions that are attempting to update overlapping sets of objects. If the transaction involves a summary field, the parent objects are locked, making these transactions especially prone to deadlocks. To debug, check your code for deadlocks and correct. Deadlocks are generally not the result of an issue with Salesforce operations.
- A timeout occurs when a transaction takes too long to complete, for example, when replacing a value in a picklist or changing a custom field definition. The timeout state is temporary. No corrective action is needed.

If an object in a batch can't be locked, the entire batch fails with this error. Errors with this status code contain the IDs of the records that couldn't be locked, when available, in the error message.

UNAVAILABLE_RECORDTYPE_EXCEPTION

The appropriate default record type could not be found.

UNDELETE_FAILED

An object could not be undeleted because it does not exist or has not been deleted.

UNKNOWN_EXCEPTION

The system encountered an internal error. Report this problem to Salesforce.

 **Note:** Do not report this exception code to Salesforce if it results from a `sendEmail()` call. The `sendEmail()` call returns this exception code when it is used to send an email to one or more recipients who have the **Email Opt Out** option selected.

UNSPECIFIED_EMAIL_ADDRESS

The specified user does not have an email address.

UNSUPPORTED_APEX_TRIGGER_OPERATION

You can't save recurring events with an Apex trigger.

UNVERIFIED_SENDER_ADDRESS

A `sendEmail()` call attempted to use an unverified email address defined in the `OrgWideEmailAddress` object.

WEBLINK_SIZE_LIMIT_EXCEEDED

The size of a WebLink URL or JavaScript code exceeds the limit.

WEBLINK_URL_INVALID

The WebLink URL has failed the URL string validation check.

WRONG_CONTROLLER_TYPE

The controller type for your Visualforce email template does not match the object type being used.

If you receive a status code not listed in the previous table, contact Customer Support.

ExtendedErrorDetails

Reserved for future use. An `ExtendedErrorDetails` element contains extra detailed information about an error. The `ExtendedErrorDetails` element has the following properties.

Name	Type	Description
<code>extendedErrorCode</code>	<code>ExtendedErrorCode</code>	A code that characterizes the extended error details.
<code>extended_error_property</code>	Varies	An extended error property that contains more information about the error. The property name and value vary based on the extended error code.

Duplicate Management Data Types

DuplicateError

Contains information about an error that occurred when an attempt was made to save a duplicate record. Use if your organization has set up duplicate rules, which are part of the Duplicate Management feature.

Fields

Field	Details
<code>duplicateResult</code>	<p>Type DuplicateResult</p> <p>Description The details of a duplicate rule and duplicate records found by the duplicate rule.</p>
<code>fields</code>	<p>Type <code>string[]</code></p> <p>Description Array of one or more field names. Identifies which fields in the object, if any, affected the error condition.</p>
<code>message</code>	<p>Type <code>string</code></p> <p>Description Error message text.</p>
<code>statusCode</code>	<p>Type StatusCode</p> <p>Description A code that characterizes the error. The full list of status codes is available in the WSDL file for your organization (see Generating the WSDL File for Your Organization).</p>

Usage

DuplicateError and its constituent objects are available to organizations that use duplicate rules.

DuplicateError is a data type of Error.

To process duplicates, loop through all the Error objects in the `errors` field on SaveResult. An Error object with a data type of DuplicateError contains information about an error that occurred when an attempt was made to save a duplicate record. To access information about the duplicates, use the `duplicateResult` field.

Java Sample

Here is a sample that shows how to see if there are any errors on the saveResult with a data type of DuplicateError. If so, duplicates were detected. See [DuplicateResult](#) for a full code sample that shows how to block users from entering duplicate leads and display an alert and a list of duplicates.

```
if (!saveResult.isSuccess()) {
    for (Error e : saveResult.getErrors()) {
        if (e instanceof DuplicateError) {
            System.out.println("Duplicate(s) Detected for lead with ID: " +
leads[i].getId());
            System.out.println("ERROR MESSAGE: " + e.getMessage());
            System.out.println("STATUS CODE: " + e.getStatusCode());
            DuplicateResult dupeResult = ((DuplicateError)e).getDuplicateResult();
            System.out.println("Found the following duplicates...");
            for (MatchResult m : dupeResult.getMatchResults()) {
                if (m.isSuccess()) {
                    System.out.println("The match rule that was triggered was " +
m.getRule());
                    for (MatchRecord mr : m.getMatchRecords()) {
                        System.out.println("Your record matched " + mr.getRecord().getId() +
" of type "
+ mr.getRecord().getType());
                        System.out.println("The match confidence is " +
mr.getMatchConfidence());
                    }
                }
            }
        }
    }
}
```

DuplicateResult

Represents the details of a duplicate rule that detected duplicate records and information about those duplicate records.

Fields

Field	Details
allowSave	Type boolean

Field	Details
	<p>Description <code>true</code> if duplicates are allowed to be saved. <code>false</code> if duplicates are not allowed to be saved.</p>
duplicateRule	<p>Type <code>string</code></p> <p>Description The name of the duplicate rule that detected duplicate records.</p>
duplicateRuleEntityType	<p>Type <code>string</code></p> <p>Description The name of the duplicate rule that detected duplicate records.</p>
errorMessage	<p>Type <code>string</code></p> <p>Description The error message configured by the administrator to warn users they are potentially creating duplicate records. This message is associated with a duplicate rule.</p>
matchResults	<p>Type MatchResult</p> <p>Description The duplicate records and related match information.</p>

Usage

DuplicateResult and its constituent objects are available to organizations that use duplicate rules.

Java Sample

Here is a sample that shows how to block users from entering duplicate leads and display an alert and a list of duplicates.

```
package com.doc.example;

import java.io.FileNotFoundException;

import com.sforce.soap.partner.*;
import com.sforce.soap.partner.Error;
import com.sforce.soap.partner.sobject.SObject;
import com.sforce.ws.ConnectionException;
import com.sforce.ws.ConnectorConfig;

public class SaveResultsWithDupeHeader {
```

```
private PartnerConnection partnerConnection = null;
static SaveResultsWithDupeHeader tester;

public static void main(String[] args) {
    tester = new SaveResultsWithDupeHeader();
    try {
        tester.demoDuplicateRuleHeader();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

/*
 * Make sure that you have an active lead duplicate rule linked to an active matching
rule. This method tries to save
 * an array of leads and inspects whether any duplicates were detected
 */
public void demoDuplicateRuleHeader() throws FileNotFoundException, ConnectionException
{
    // Create the configuration for the partner connection
    ConnectorConfig config = new ConnectorConfig();
    config.setUsername("user@domain.com");
    config.setPassword("secret");
    config.setAuthEndpoint("authEndPoint");
    config.setTraceFile("traceLogs.txt");
    config.setTraceMessage(true);
    config.setPrettyPrintXml(true);

    // Initialize the connection
    partnerConnection = new PartnerConnection(config);

    // Get the leads that have to be saved
    SObject[] leads = getLeadsForInsertOrUpdate();

    /* Set a duplicate rule header to return a result if duplicates are detected
     * This sets the allowSave, includeRecordDetails, and runAsCurrentUser flags to
true
     */
    partnerConnection.setDuplicateRuleHeader(true, true, true);

    // Save the leads
    SaveResult[] saveResults = partnerConnection.create(leads);

    // Check the results to see if duplicates were detected
    for (int i = 0; i < leads.length; i++) {
        SaveResult saveResult = saveResults[i];
        if (!saveResult.isSuccess()) {
            for (Error e : saveResult.getErrors()) {
                // See if there are any errors on the saveResult with a data type of
DuplicateError
                if (e instanceof DuplicateError) {
                    System.out.println("Duplicate(s) Detected for lead with ID: " +
leads[i].getId());
                    System.out.println("ERROR MESSAGE: " + e.getMessage());
                }
            }
        }
    }
}
```

```

        System.out.println("STATUS CODE: " + e.getStatusCode());
        DuplicateResult dupeResult =
((DuplicateError)e).getDuplicateResult();
        System.out.println("Found the following duplicates...");
        for (MatchResult m : dupeResult.getMatchResults()) {
            if (m.isSuccess()) {
                System.out.println("The match rule that was triggered was
" + m.getRule());
                for (MatchRecord mr : m.getMatchRecords()) {
                    System.out.println("Your record matched " +
mr.getRecord().getId() + " of type "
+ mr.getRecord().getType());
                    System.out.println("The match confidence is " +
mr.getMatchConfidence());
                    for (FieldDiff f : mr.getFieldDiffs()) {
                        System.out.println("For field " + f.getName() + "
field difference is "
+ f.getDifference().name());
                    }
                }
            }
        }
    }

    // Clear the duplicate rule header
    partnerConnection.clearDuplicateRuleHeader();
}

/**
 * The assumption here is that this method is retrieving leads from either UI, a data
source, etc
 */
private SObject[] getLeadsForInsertOrUpdate() {
    return new SObject[0];
}
}

```

MatchResult

Represents the duplicate results for a matching rule.

Fields

Field	Details
errors	Type
	Error

Field	Details
	<p>Description Errors that occurred during matching for the matching rule.</p>
entityType	<p>Type string</p> <p>Description The entity type of the matching rule.</p>
matchEngine	<p>Type string</p> <p>Description The match engine for the matching rule.</p>
matchRecords	<p>Type MatchRecord[]</p> <p>Description Information about the duplicates detected by the matching rule.</p>
rule	<p>Type string</p> <p>Description The developer name of the matching rule that detected duplicates.</p>
size	<p>Type int</p> <p>Description The number of duplicates detected by the matching rule.</p>
success	<p>Type boolean</p> <p>Description true if the matching rule successfully ran. false if there's an error with the matching rule.</p>

Usage

MatchResult and its constituent objects are available to organizations that use duplicate rules.

There is 1 MatchResult for each saved record that has duplicates. The MatchResult contains all duplicates for that saved record.

Java Sample

Here is a sample that shows how to display the ID and type of all duplicates detected when leads are saved. See [DuplicateResult](#) for a full code sample that shows how to block users from entering duplicate leads and display an alert and a list of duplicates.

```
for (MatchResult m : dupeResult.getMatchResults()) {
    if (m.isSuccess()) {
        System.out.println("The match rule that was triggered was " + m.getRule());
        for (MatchRecord mr : m.getMatchRecords()) {
            System.out.println("Your record matched " + mr.getRecord().getId() + " of type
"
                + mr.getRecord().getType());
            System.out.println("The match confidence is " + mr.getMatchConfidence());
        }
    }
}
```

MatchRecord

Represents a duplicate record detected by a matching rule.

Fields

Field	Details
additionalInformation	<p>Type AdditionalInformationMap</p> <p>Description Other information about matched records.</p>
fieldDiffs	<p>Type FieldDiff[]</p> <p>Description Matching rule fields and how each field value compares for the duplicate and its matching record.</p>
matchConfidence	<p>Type double</p> <p>Description The ranking of how similar a matched record's data is to the data in your request. Must be equal to or greater than the value of the <code>minMatchConfidence</code> specified in your request. Returns -1 if unused.</p>
record	<p>Type sObject</p> <p>Description The fields and field values for the duplicate record.</p>

Usage

MatchRecord and its constituent objects are available to organizations that use duplicate rules.

Each MatchRecord represents a duplicate detected when a record is saved. There can be multiple MatchRecord objects for a saved record if multiple duplicates are detected.

Java Sample

Here is a sample that shows how to display the ID and type of all duplicates detected when a lead is saved. See [DuplicateResult](#) for a full code sample that shows how to block users from entering duplicate leads and display an alert and a list of duplicates.

```
for (MatchRecord mr : m.getMatchRecords()) {
    System.out.println("Your record matched " + mr.getRecord().getId() + " of type "
        + mr.getRecord().getType());
    System.out.println("The match confidence is " + mr.getMatchConfidence());
}
```

FieldDiff

Represents the name of a matching rule field and how the values of the field compare for the duplicate and its matching record.

Fields

Field	Details
difference	<p>Type differenceType</p> <p>Description How the values of the matching rule field compare for the duplicate and its matching record. Possible values include:</p> <ul style="list-style-type: none"> • Same: Indicates the field values match exactly. • Different: Indicates that the field values do not match. • Null: Indicates that the field values are a match because both values are blank.
name	<p>Type string</p> <p>Description The name of a field on a matching rule that detected duplicates.</p>

Java Sample

Here is a sample that shows how to display the matching rule field differences when a duplicate is detected. See [DuplicateResult](#) for a full code sample that shows how to block users from entering duplicate leads and display an alert and a list of duplicates.

```
for (FieldDiff f : mr.getFieldDiffs()) {
    System.out.println("For field " + f.getName() + " field difference is "
```

```
+ f.getDifference().name());
}
```

AdditionalInformationMap

Represents other information, if any, about matched records.

Fields

Field	Details	
	Type	Description
name	string	The name of the element.
value	string	The value of the element.

System Fields

The following fields are read-only fields found on most objects. These fields are automatically updated during API operations. For example, the ID field is automatically generated during a create operation and the LastModifiedDate is automatically updated when a user modifies a record.

Field	Field Type	Description
Id	ID	Globally unique string that identifies a record. For information on IDs, see ID Field Type . Because this field exists in every object, it is not listed in the field table for each object. Id fields have Defaulted on create and Filter access.
IsDeleted	boolean	Indicates whether the record has been moved to the Recycle Bin (true) or not (false). Because this field does not appear in all objects, it is listed in the field table for each object.

Audit Fields

CreatedBy	reference	ID of the User who created this record. CreatedBy fields have Defaulted on create and Filter access.
CreatedDate	dateTime	Date and time when this record was created. CreatedDate fields have Defaulted on create and Filter access.
LastModifiedBy	reference	ID of the User who last updated this record. LastModifiedBy fields have Defaulted on create and Filter access.

Field	Field Type	Description
LastModifiedDate	dateTime	Date and time when a user last modified this record. <code>LastModifiedDate</code> fields have Defaulted on create and Filter access.
SystemModstamp	dateTime	Date and time when a user or automated process (such as a trigger) last modified this record. <code>SystemModstamp</code> fields have Defaulted on create and Filter access.

If you import data into Salesforce and want to retain the audit field values of the source system, you can set the values for audit fields on the following objects: Account, ArticleVersion, Attachment, CampaignMember, Case, CaseComment, Contact, ContentVersion, Contract, Event, Idea, IdeaComment, Lead, Opportunity, Question, Task, Vote, and custom objects. The only audit field you cannot set a value for is `SystemModstamp`.

1. From Setup, enter `User Interface` in the Quick Find box, then select **User Interface** under Customize.
2. Under Setup, select **Enable “Set Audit Fields upon Record Creation” and “Update Records with Inactive Owners” User Permissions**.
3. In the permission set or profile that you want to set audit fields with, enable the permission **Set Audit Fields upon Record Creation**.
4. Using the API, create a record and set its audit fields.

Not all standard objects have all audit fields. Check the Enterprise WSDL to verify which audit fields are available for a given object.

Parent Reference Fields

If an object has a relationship to a parent object, two fields are added.

- `Parent_Name` contains the object name of the parent. For example, Case has a `Contact` field that contains a reference to the contact parent of the case.
- `Parent_NameId` contains the ID of the parent. For example, Case has a `ContactId` field that refers to the contact parent of the case. This field is used in SOQL relationship queries such as the following:

```
SELECT Case.ContactId, Case.Contact.Name FROM Case
```

Even if the object can parent itself, these fields occur. For example, the Campaign object has a `Campaign` and `CampaignId` field for referencing the parent Campaign.

Required Fields

Required fields must have a non-`null` value. This rule affects the create and update calls:

- In a create call, the system automatically populates the data for certain required fields (such as system fields and the object ID fields). Similarly, if a required field has a default value (its `defaultedOnCreate` attribute is set to `true`, as described in Field in a describe result), then the system implicitly assigns a value for this field when the object is created, even if a value for this field is not explicitly passed in on the create call. For all other required fields, such as ID fields that are analogous to foreign keys in SQL, a client application must explicitly assign a value when the object is created (it cannot be `null`).
- In updates, a required field cannot be set to `null`, and many required fields can't be changed.

Any field not specified as required in the object description is optional, that is, it can be `null` when updated or created.

Some required fields for some objects require special handling.

Frequently-Occurring Fields

In addition to system fields, the following fields are found on many objects:

- `OwnerId`
- `RecordTypeId`
- `CurrencyIsoCode`

OwnerId

Objects have an `OwnerId` field that is a reference to the user who owns that object. Ownership is an important concept that affects the security model and has other implications throughout the system. Any user can query the owner field for any record they can access. However, setting the `OwnerId` field has the following limitations:

- For most users and most objects, this field can't be set directly upon insert. It is implicitly set to the current user when inserting an object.
- When creating or updating a Case or Lead, a client application (that is logged in with sufficient permissions to transfer a record) can set this field to any valid User in the organization or to any valid queue of the appropriate type in the organization.
- Updating this field via the API changes only the owner of that record. The change of ownership does not cascade to associated records as it does when you transfer record ownership in the Salesforce user interface.
- Updating this field on an account deletes the existing sharing information and reapplys the organization-wide sharing defaults and sharing rules.
- To update the `OwnerId` field, the user must have the "Transfer Record" permission and Read access to the new owner.

In API version 12.0 and later, if your organization has set up opportunity teams, `OwnerId` fields behave the same for Account and Opportunity objects as for other objects. That is, if you update the `OwnerId` field in either object, any AccountShare or OpportunityShare records with `RowCause` set to Sales Team are kept. In API version 11.0 and earlier, the sharing records are deleted.

RecordTypeId

Record types are used to offer different business processes and subsets of picklist values to different User records based on their Profile settings. (In addition, person accounts use record types to manage a number of additional elements.)

Record types are configured in the user interface or by creating, editing, or deleting the `RecordType` object in the API. Retrieve the list of valid record type IDs (String) for an object by querying the `RecordType` object.

The `RecordTypeId` field in an object contains the ID of the `RecordType` record that is associated with a standard or custom object. You can create or update this field.

-  **Note:** You can't create or update the `RecordTypeId` field on the `CampaignMember` records. Set the `CampaignMember` record type using the `CampaignMemberRecordTypeId` field on `Campaign`.

When specified in a create or update call, the record type ID (String) must refer to a valid record type for that object.

-  **Note:** The `RecordTypeId` field is in your organization's WSDL only if at least one record type is configured for your organization in the Salesforce user interface.

CurrencyIsoCode

For organizations that have multicurrency enabled, the `CurrencyIsoCode` field contains the string representation of the currency ISO code associated with currency values in the object. Note that the `User` object also has a `DefaultCurrencyIsoCode` field,

which is the default currency for that user. For example, a user in France could have a `DefaultCurrencyIsoCode` set to Euros, and that would be their default currency in the application. However, the User object could have currency custom fields stored in a different currency, that will correspond to the organization currency at the time the user record is created.

API Field Properties

Fields on objects represent the details of each object and are analogous to columns in a database table. Each field on each object has one or more of the following properties:

Property	Description
Autonumber	The API creates an autonumber.
Create	Value for the field can be specified during create using the API.
Defaulted on create	When created, a default value is supplied if no other value is specified.
Delete	Value for the field can be deleted using the API.
Filter	Can be used as filter criteria in a SOQL query FROM or WHERE clause.
Group	Can be included in the GROUP BY clause of a SOQL query (<code>true</code>) or not (<code>false</code>). Available in API version 18.0 and later.
idLookup	Can be used to specify a record in an upsert call. The <code>Id</code> field of each object has this property and some <code>Name</code> fields. There are exceptions, so check for the property in any object you wish to upsert.
Nillable	The field can contain a null value.
Query	The field can be queried with SOQL using the API.
Restricted picklist	A picklist whose values are restricted to those values defined by a Salesforce admin. Users can't load unapproved values through the API.
Retrieve	Value of the field can be retrieved using the API.
Sort	Indicates whether a query can sort on this field (<code>true</code>) or not (<code>false</code>).
Update	Can be updated using the API.

Relationships Among Objects

Relationships associate objects with other objects. For example, a relationship can link a custom object to standard object in a related list, such as linking a custom object called Bugs to cases to track product defects associated with customer cases. To view the parent and child relationships among standard objects, see the ERD diagrams in [Data Model](#).



Note:

- You can use parent-child relationships in SOQL queries. For more information, see Relationship Queries in the [Salesforce SOQL and SOSL Reference Guide](#).
- Only lookup, external lookup, and indirect lookup relationships are available for external objects. No other relationship types are supported. See "External Object Relationships" in the Salesforce Help.

You can define different types of relationships by creating custom relationship fields on an object. The differences between relationship types include how they handle data deletion, record ownership, security, and required fields in page layouts:

- **Master-Detail (1:n)** — A parent-child relationship in which the master object controls certain behaviors of the detail object:
 - When a record of the master object is deleted, its related detail records are also deleted.
 - The `Owner` field on the detail object is not available and is automatically set to the owner of its associated master record. Custom objects on the detail side of a master-detail relationship cannot have sharing rules, manual sharing, or queues, as these require the `Owner` field.
 - The detail record inherits the sharing and security settings of its master record.
 - The master-detail relationship field is required on the page layout of the detail record.
 - By default, records can't be reparented in master-detail relationships. Administrators can, however, allow child records in master-detail relationships on custom objects to be reparented to different parent records by selecting the `Allow reparenting` option in the master-detail relationship definition.

You can define master-detail relationships between custom objects or between a custom object and a standard object. However, the standard object cannot be on the detail side of a relationship with a custom object. In addition, you cannot create a master-detail relationship in which the User or Lead objects are the master.

When you define a master-detail relationship, the custom object on which you are working is the detail side. Its data can appear as a custom related list on page layouts for the other object.

- **Many-to-many** — You can use master-detail relationships to model *many-to-many* relationships between any two objects. A many-to-many relationship allows each record of one object to be linked to multiple records from another object and vice versa. For example, you may have a custom object called "Bug" that relates to the standard case object such that a bug could be related to multiple cases and a case could also be related to multiple bugs. To create a many-to-many relationship, simply create a custom junction object with two master-detail relationship fields, each linking to the objects you want to relate. See the Salesforce online help for details.

Custom objects with two master-detail relationships are supported in API version 11 and later.

Starting in API version 34.0, the `JunctionIdList` field type lets you manipulate the many-to-many relationship of an entity directly. You no longer need to manipulate underlying junction entity records. `JunctionIdList` fields can be queried and updated like any other field on the entity. Queries or updates to `JunctionIdList` fields act as queries or updates to the underlying junction entity records. Fields of type `JunctionIdList` appear in the WSDL as an unbounded array of type ID. `JunctionIdList` is implemented in the `Task` and `Event` objects.

- **Lookup (1:n)** — This type of relationship links two objects together, but has no effect on deletion or security. Unlike master-detail fields, lookup fields are not automatically required. When you define a lookup relationship, data from one object can appear as a custom related list on page layouts for the other object. See the Salesforce online help for details.

To create relationships, use the user interface or Salesforce Metadata API.

Relabeling Fields and Tabs and the API

The user interface allows you to change the labels on some fields and tabs. Although you cannot relabel fields or tabs using the API, you can retrieve the current values. To do so, issue a `describeSObjects()` call and inspect the `label` field of the returned `DescribeSObjectResult`.

Tooling API Objects in the Enterprise WSDL

Some objects used by the Tooling API are included in the Enterprise and Partner WSDL. Use the objects from these WSDLs instead of the Tooling WSDL if you need field-level security.

The following Tooling API objects are exposed in the Enterprise and Partner WSDL.

- `DataType`
- `EntityDefinition`
- `EntityParticle`
- `FieldDefinition`
- `PicklistValueInfo`
- `Publisher`
- `SearchLayout`
- `Service`
- `ServiceDataType` (Reserved for future use.)
- `ServiceFieldDataType` (Unavailable in version 35.0 and later. Do not use.)
- `RelationshipDomain`
- `RelationshipInfo`
- `UserEntityAccess`
- `UserFieldAccess`
- `WsdlDataType` (Reserved for future use.)
- `XmlSchema` (Reserved for future use.)

For more information, use the *Tooling API Developer's Guide*.

Force.com AppExchange Object Prefixes and the API

If you have an unmanaged package and a managed package version becomes available, the API names of custom fields, custom objects, and Scontrol objects in the package change. A namespace prefix is added to each component to make it unique: `name__c` becomes `prefix__name__c`. To move from an unmanaged package to a managed package version of the same application, export your data, uninstall the old package, and install the new package. Then review the name changes and import your data with the relevant mapping. For details, see the [ISVforce Guide](#).

Custom Objects

In the user interface, you can extend your organization's data by defining custom objects. Custom objects are custom database tables that allow you to store information unique to your organization. For custom objects, the `custom` flag—a Boolean field in the describe results—is `true`.

Client applications with sufficient permissions can invoke API calls on existing custom objects. You can create custom objects with the user interface, or by using the metadata WSDL with a client application or using the Force.com IDE. For more information about using the metadata WSDL to create custom objects, see the [Force.com Metadata API Developer's Guide](#). For more information about the Force.com IDE, see [Salesforce Developers](#).

Use the following topics to understand how the API interacts with custom objects and fields:

- [Naming Conventions for Custom Objects](#)
- [Relationships Among Custom Objects](#)
- [Audit Fields for Custom Objects](#)
- [Sharing and Custom Objects](#)
- [Required Fields in Custom Objects](#)
- [Managed Packages and API Names](#)

Naming Conventions for Custom Objects

Your Salesforce administrator defines an associated name field for each custom object during setup. Custom objects must have unique names within your organization.

In the API, the names of custom objects include a suffix of two underscores followed by a lowercase “c”. For example, a custom object labeled “Issue” in the Salesforce user interface is `Issue__c` in that organization’s WSDL.

Relationships change the naming convention. See [Relationships Among Custom Objects](#) for more information.

For a custom object record to appear in the Salesforce user interface, its name field must be populated. If you use the API to create a custom object record that doesn’t have a name, the record’s ID is used as its name.

Relationships Among Custom Objects

Custom objects relate to other objects and behave just like standard objects, as described in [Relationships Among Objects](#). For example, cascading deletes are supported in custom objects in a Master-Detail relationship.

Custom objects require special treatment so that they can participate in Relationship Queries. For the relationship field name of a custom object, `_r` is appended to the name to create the ID. Also, `_c` is appended to the name to create the parent object pointer. For example, if the relationship field name is `MyRel`, the name of the ID becomes `MyRelId__r`, the parent object pointer becomes `MyRel__c`, and the relationship name is `MyRel__r`. For more information, see [Understanding Relationship Names, Custom Objects, and Custom Fields](#) in the [Salesforce SOQL and SOSL Reference Guide](#).

The following table summarizes whether a standard object can be:

- The master in a master-detail relationship with a custom object. Master-detail relationships involve cascading deletes and sharing rules that the parent controls.
- The lookup in a lookup relationship on a custom object. In other words, whether a custom object can have a lookup to the standard object.
- Extended with custom fields.

Standard Object	Master-Detail	Lookup	Custom Fields
Account	Yes	Yes	Yes
Campaign	Yes	Yes	Yes
Case	Yes	Yes	Yes
Contact	Yes	Yes	Yes
Contract	Yes	Yes	Yes
Event	No	No	Yes
Lead	No	No	Yes

Standard Object	Master-Detail	Lookup	Custom Fields
Opportunity	Yes	Yes	Yes
Product2	No	Yes	Yes
Solution	Yes	Yes	Yes
Task	No	No	Yes
User	No	Yes	Yes

Custom objects can also have many-to-many relationships with other custom objects or standard objects. A many-to-many relationship allows each record of one object to be linked to multiple records from another object and vice versa. For more information, see [Relationships Among Objects](#).

Audit Fields for Custom Objects

Custom objects can have the same audit fields as standard objects. When you create a custom object, the four audit fields, `CreatedById`, `CreatedDate`, `LastModifiedById`, and `LastModifiedDate`, are created and populated for the object. These fields are read only. If you import data into Salesforce custom objects and want to retain the audit field values from the source system, you can set the values when you create the custom objects. The only audit field you cannot set a value for is `systemModstamp`. Your organization must be API enabled, and you must have the "Modify All Data" permission.

1. From Setup, enter `User Interface` in the Quick Find box, then select **User Interface** under Customize.
2. Under Setup, select **Enable "Set Audit Fields upon Record Creation" and "Update Records with Inactive Owners" User Permissions**.
3. In the permission set or profile that you want to set audit fields with, enable the permission **Set Audit Fields upon Record Creation**.
4. Using the API, create a record and set its audit fields.

Note these restrictions:

- `CreatedDate` can't be greater than the `LastModifiedDate`.
- You can't set any date field to be greater than the current time.

For more information about audit fields, see [System Fields](#).

Sharing and Custom Objects

A sharing rule object is created for each custom object that does not have a master-detail relationship to another object. They are similar to standard object sharing rules, for example `AccountOwnerSharingRule`. If the user creating the custom object has the "Manage Sharing" permission, a sharing rule object is automatically created for it.

Apex sharing reasons can be retrieved describing the custom object's sharing object, and examining the information in the `rowCause` field. The name of a sharing object for each custom object is of the form: `MyObjectName__Share`, similar to `AccountShare` and other standard object sharing objects.

Tags and Custom Objects

When a custom object is created, a Tag object related to it is also created. These object names are of the form: `MyObjectName__Tag`, similar to `AccountTag` and other standard object tag objects.

Required Fields in Custom Objects

In the user interface, you can mark a custom field as required, and this rule is also enforced in the API. Each custom field has a `isRequired` field, with a data type boolean. The default value is `false`. If set to `true`, each request supplies a value (or leaves the current value) to this field. Otherwise, the request fails. When the value is set to `true`, the next time the field is edited or created, the validation applies. If no value is supplied or default value specified, the request fails.

To edit the `isRequired` field, you must log in as a user with the “Customize Application” permission.

If you change a custom object field to be required in an existing client application or integration, be sure that a value is supplied for that field. For example, if the custom picklist field `Education Level` on the contact object is required, supply a default value for that custom field. If a required field does not have a specified or default value, an error with the status code `REQUIRED_FIELD_MISSING` is returned.

Managed Packages and API Names

If you have an unmanaged package and a managed package version becomes available, the API names of custom fields, custom objects, and Scontrol objects in the package change. A namespace prefix is added to each component to make it unique: `name__c` becomes `prefix__name__c`. To move from an unmanaged package to a managed package version of the same application, export your data, uninstall the old package, and install the new package. Then review the name changes and import your data with the relevant mapping. For details, see the [ISVforce Guide](#).

External Objects

External objects are supported in API version 32.0 and later. External objects are similar to custom objects, but external object record data is stored outside your Salesforce organization. For example, perhaps you have data that's stored on premises in an enterprise resource planning (ERP) system. Instead of copying the data into your org, you can use external objects to access the data in real time via web service callouts.

External objects are available with Salesforce Connect and Files Connect. Each external object is associated with an external data source definition in your Salesforce organization.

An external data source specifies how to access an external system. Salesforce Connect uses external data sources to access data that's stored outside your Salesforce organization. Files Connect uses external data sources to access third-party content systems. External data sources have associated external objects, which your users and the Force.com platform use to interact with the external data and content.

By accessing record data on demand, external objects always reflect the current state of the external data. You don't have to manage a copy of that data in Salesforce, so you're not wasting storage and resources keeping data in sync.

External objects are best used when you have a large amount of data that you can't or don't want to store in your Salesforce organization, and you only need to use a small amount of data at any one time.

See “Define External Objects” in the Salesforce Help for how to create and modify external objects.

Naming Conventions for External Objects

Object names must be unique across all standard, custom, and external objects in the org.

In the API, the names of external objects are identified by a suffix of two underscores immediately followed by a lowercase “x” character. For example, an external object named “ExtraLogInfo” in the Salesforce user interface is seen as `ExtraLogInfo__x` in that organization's WSDL.

We recommend that you make object labels unique across all standard, custom, and external objects in the org.

External Object Relationships

External objects support standard lookup relationships, which use the 18-character Salesforce record IDs to associate related records with each other. However, data that's stored outside your Salesforce org often doesn't contain those record IDs. Therefore, two special types of lookup relationships are available for external objects: external lookups and indirect lookups. See "External Object Relationships" in the Salesforce Help for details.

Feature Support for External Objects

Most of the Salesforce features that support custom objects also support external objects. However, there are exceptions, and some features have special limitations and considerations for external objects. See the following topics in the Salesforce Help.

- Salesforce Compatibility Considerations for Salesforce Connect—All Adapters
- Considerations for Salesforce Connect—All Adapters

Salesforce Connect Adapters

Salesforce Connect uses a protocol-specific adapter to connect to an external system and access its data. This table describes the available adapters.

Salesforce Connect Adapter	Description	Where to Find Callout Limits
Cross-org	Uses the Force.com REST API to access data that's stored in other Salesforce orgs.	No callout limits. However, each callout counts toward the API usage limits of the provider org. Salesforce Help: API Usage Considerations for Salesforce Connect—Cross-Org Adapter Salesforce Limits Quick Reference Guide: API Requests Limits
OData 2.0 OData 4.0	Uses Open Data Protocol to access data that's stored outside Salesforce. The external data must be exposed via OData producers.	Salesforce Help: General Limits for Salesforce Connect—OData 2.0 and 4.0 Adapters
Custom adapter created via Apex	You use the Apex Connector Framework to develop your own custom adapter when the other available adapters aren't suitable for your needs. A custom adapter can obtain data from anywhere. For example, some data can be retrieved from anywhere in the Internet via callouts, while other data can be manipulated or even generated programmatically.	Apex Developer Guide: Callout Limits and Limitations Apex Developer Guide: Execution Governors and Limits

For more information about Salesforce Connect, see "Salesforce Connect" in the Salesforce Help.

For details on using the Apex Connector Framework, see "Salesforce Connect" and "DataSource Namespace" in the [Force.com Apex Code Developer's Guide](#).

CHAPTER 3 API Call Basics

In this chapter ...

- [Characteristics of API Calls](#)
- [Factors that Affect Data Access](#)
- [Package Version Settings](#)

API calls represent specific operations that your client applications can invoke at runtime to perform tasks, for example:

- Query data in your organization.
- Add, update, and delete data.
- Obtain metadata about your data.
- Run utilities to perform administration tasks.

Using your development environment, you can construct Web service client applications that use standard Web service protocols to programmatically:

- Log in to the login server (`login()`) and receive authentication information to be used for subsequent calls
- Query your organization's information (`query()`, `queryAll()`, `queryMore()`, and `retrieve()` calls)
- Perform text searches across your organization's information (`search()` call)
- Create, update, and delete data (`create()`, `merge()`, `update()`, `upsert()`, `delete()`, and `undelete()` calls)
- Perform administrative tasks, such as retrieving user information (`getUserInfo()` call), changing passwords (`setPassword()` and `resetPassword()` calls), and getting the system time (`getServerTimestamp()` call)
- Replicate data locally (`getDeleted()` and `getUpdated()` calls)
- Obtain and navigate metadata about your organization's data (`describeGlobal()`, `describeSObject()`, `describeSObjects()`, `describeLayout()`, and `describeTabs()` calls)
- Work with approval processes (`process()`)
- Return category groups and categories from your organization (`describeDataCategoryGroups()` and `describeDataCategoryGroupStructures()`).

See [Core Calls](#), [Describe Calls](#), and [Utility Calls](#) for complete details about each call.

Characteristics of API Calls

All API calls are:

- **Service Requests and Responses**—Your client application prepares and submits a service request to the Force.com Web Service via the API, the Force.com Web Service processes the request and returns a response, and the client application handles the response.
- **Synchronous**—Once the API call is invoked, your client application waits until it receives a response from the service. Asynchronous calls are not supported.
- **Committed Automatically Versus Rollback on Error**—By default, every operation that writes to a Salesforce object is committed automatically. This is analogous to the AUTOCOMMIT setting in SQL. For `create()`, `update()`, and `delete()` calls that attempt to write to multiple records for an object, the write operation for each record is treated as a separate transaction. For example, if a client application attempts to create two new accounts, they're created using mutually exclusive insert operations that succeed or fail individually, not as a group.

For API version 20.0 and later, there is an `AllOrNoneHeader` header that allows a call to roll back all changes unless all records are processed successfully. This header is supported by the `create()`, `delete()`, `undelete()`, `update()`, and `upsert()` calls.

-  **Note:** The default behavior means that client applications may need to handle some failures: for example, if you create an opportunity that has shipments (a custom object), and the opportunity line item gets created but the shipment creation fails, if your business rules required all opportunities be created with shipment, your client application would need to roll back the opportunity creation. The easiest way to do this is to use `AllOrNoneHeader`.

Factors that Affect Data Access

When using the API, the following factors affect access to your organization's data:

Access

Your organization must be enabled for API access.

Objects may not be available until you contact Salesforce and request access. For example Territory is visible only if territory management has been enabled in the application. Such requirements are in the "Usage" section for each object.

Sometimes a feature must be used once before objects related to it can be accessed with the API. For example, the `recordTypeIds` is available only after at least one record type has been created for your organization in the user interface.

To investigate data access issues, you can start by inspecting the WSDL:

- **Enterprise WSDL:** The generated enterprise WSDL file contains all of the objects that are available to your organization. By using the API, a client application can access objects that are defined in your enterprise WSDL file.
- **Partner WSDL:** When using the generated partner WSDL file, a client application can access objects that are returned in the `describeGlobal()` call.

Object-Level and Field-Level Security

The API respects object-level and field-level security configured in the user interface. You can access objects and fields only if the logged-in user's permissions and access settings allow such access. For example, fields that are not visible to a given user are not returned in a `query()` or `describeSObjects()` call. Similarly, read-only fields can't be updated.

User Permissions

A user attempting to access the API must have the permission "API Enabled" selected. It's selected by default.

Your client application logs in as a user called a *logged-in* user. The logged-in user's permissions grant or deny access to specific objects and fields in your organization:

- **Read**—Users can only view objects of this type.
- **Create**—Users can read and create objects of this type.
- **Edit**—Users can read and update objects of this type.
- **Delete**—Users can read, edit, and delete objects of this type.

User permissions do not affect field-level security. If field-level security specifies that a field is hidden, users with “Read” on that object can view only those fields that are not hidden on the record. In addition, users with “Read” on an object can view only those records that sharing settings allow. The one exception is the “Edit Read Only Fields” permission, which gives users the ability to edit fields marked as read only via field-level security.

Sharing

For most API calls, data that is outside of the logged-in user’s sharing model is not returned. Users are granted the most permissive access that is available to them, either through organization-wide defaults or manual record sharing, just as in the application.

User Permissions that Override Sharing

- **View All**—Users can view all records associated with this object, regardless of sharing settings.
- **Modify All**—Users can read, edit, delete, transfer, and approve all records associated with this object, regardless of sharing settings.
- **Modify All Data**—users can read, edit, delete, transfer, and approve all records regardless of sharing settings. This permission is not an object-level permission, unlike “View All” and “Modify All.”

To protect the security of your data, give the logged-in user only the permissions needed to successfully execute all the calls made by the application. For large integration applications, “Modify All Data” may speed up call response times. If you are loading a large number of records, use the [Bulk API](#) instead.

Related Objects

Some objects depend on other objects for permission. For example, AccountTeamMember follows sharing on the associated permission-assigned object such as the Account record. Similarly, a Partner depends on the permissions in the associated .

Ownership changes to a record do not automatically cascade to related records. For example, if ownership changes for a given Account, ownership does not then automatically change for any Contract associated with that Account—each ownership change must be made separately and explicitly by the client application.

Object Properties

To create an object with the `create()` call, the object’s `createable` attribute must be set to `true`. To determine what operations are allowed on a given object, your client application can invoke the `describeSObjects()` call on the object and inspect the properties in the `DescribeSObjectResult`.

 **Note:** `replicable` allows `getUpdated()` and `getDeleted()` calls.

Page Layouts and Record Types

Requirements defined in the Salesforce user interface for page layouts and record types are not enforced by the API:

- Page layouts can specify whether a given field is required, but the API does not enforce such layout-specific field restrictions or validations in `create()` and `update()` calls. It’s up to the client application to enforce any such constraints, if applicable.
- Record types can control which picklist values can be chosen in a given record and which page layouts users with different profiles can see. However, such rules that are configured and enforced in the user interface are not enforced in the API. For example, the API does not validate whether the value in a picklist field is allowed per any record type restrictions associated with the profile of the logged-in user. Similarly, the API does not prevent a client application from adding data to a particular field simply because that field does not appear in a layout associated with the profile of the logged-in user.

Referential Integrity

To ensure referential integrity, the API forces or prevents certain behaviors:

- ID values in [reference fields](#) are validated in `create()` and `update()` calls.

- If a client application deletes a record, then its children are automatically deleted as part of the call if the `cascadeDelete` property on ChildRelationship for that child has a value of `true`. For example, if a client application deletes an Opportunity, then any associated OpportunityLineItem records are also deleted. However, if an OpportunityLineItem is not deletable or is currently being used, then deletion of the parent Opportunity fails. For example, if a client application deletes an Invoice_Statement, then any associated Line_Item records are also deleted. However, if a Line_Item is not deletable or is currently being used, then deletion of the parent Invoice_Statement fails. Use `DescribeSObjectResult` to view the `ChildRelationship` value if you want to be sure what will be deleted.

There are certain exceptions that prevent the execution of a `cascadeDelete`. For example, you can't delete an account if it has associated cases, if it has related opportunities that are owned by other users, or if associated contacts are enabled for the Customer Portal. In addition, if you attempt to delete an account that has closed/won opportunities owned by you or has active contracts, then the delete request for that record will fail.

Package Version Settings

When your API client is referencing components in managed packages, you can specify the version of each installed package that you want to reference for your integration. This allows your API client to continue to function with specific, known behavior even when you install subsequent versions of a package. You can use the [PackageVersionHeader](#) SOAP header to set different package versions for different calls, if necessary.

A package version is a number that identifies the set of components uploaded in a package. The version number has the format `majorNumber.minorNumber.patchNumber` (for example, 2.1.3). The major and minor numbers increase to a chosen value during every major release. The `patchNumber` is generated and updated only for a patch release. Publishers can use package versions to evolve the components in their managed packages gracefully by releasing subsequent package versions without breaking existing customer integrations using the package.

Default package versions for API calls provide fallback settings if package versions are not provided by an API call. Many API clients do not include package version information, so the default settings maintain existing behavior for these clients.

You can specify the default package versions for enterprise API and partner API calls. The enterprise WSDL is for customers who want to build an integration with their Salesforce organization only. It is strongly typed, which means that calls operate on objects and fields with specific data types, such as `int` and `string`. The partner WSDL is for customers, partners, and ISVs who want to build an integration that can work across multiple Salesforce organizations, regardless of their custom objects or fields. It is loosely typed, which means that calls operate on name-value pairs of field names and values instead of specific data types.

You must associate the enterprise WSDL with specific package versions to maintain existing behavior for clients. There are options for setting the package version bindings for an API call from client applications using either the enterprise or partner WSDL. The package version information for API calls issued from a client application based on the enterprise WSDL is determined by the first match in the following settings.

1. The `PackageVersionHeader` SOAP header.
2. The SOAP endpoint contains a URL with a format of `serverName/services/Soap/c/api_version/ID` where `api_version` is the version of the API, such as 38.0, and `ID` encodes your package version selections when the enterprise WSDL was generated.
3. The default enterprise package version settings.

The partner WSDL is more flexible as it is used for integration with multiple organizations. If you choose the Not Specified option for a package version when configuring the default partner package versions, the behavior is defined by the latest installed package version. This means that behavior of package components, such as an Apex trigger, could change when a package is upgraded and that change would immediately impact the integration. Subscribers may want to select a specific version for an installed package for all partner API calls from client applications to ensure that subsequent installations of package versions do not affect their existing integrations.

The package version information for partner API calls is determined by the first match in the following settings.

1. The PackageVersionHeader SOAP header.
2. An API call from a Visualforce page uses the package versions set for the Visualforce page.
3. The default partner package version settings.

To configure default package versions for API calls:

1. From Setup, enter **API** in the Quick Find box, then select **API**.
2. Click **Configure Enterprise Package Version Settings** or **Configure Partner Package Version Settings**. These links are only available if you have at least one managed package installed in your organization.
3. Select a **Package Version** for each of your installed managed packages. If you are unsure which package version to select, you should leave the default selection.
4. Click **Save**.



Note: Installing a new version of a package in your organization does not affect the current default settings.

CHAPTER 4 Error Handling

In this chapter ...

- [Error Handling for Session Expiration](#)
- [More About Error Handling](#)

The API calls return error data that your client application can use to identify and resolve runtime errors. If an error occurs during the invocation of most API calls, then the API provides the following types of error handling:

- For errors resulting from badly formed messages, failed authentication, or similar problems, the API returns a SOAP fault message with an associated [ExceptionCode](#).
- For most calls, if the error occurs because of a problem specific to the query, the API returns an [Error](#). For example, if a `create()` request contains more than 200 objects.

Error Handling for Session Expiration

When you sign on via the [login\(\)](#) call, a new client session begins and a corresponding unique session ID is generated. Sessions expire automatically after a predetermined length of inactivity, which can be configured in Salesforce from Setup by clicking **Security Controls**. The default is 120 minutes (two hours). If you make an API call, the inactivity timer is reset to zero.

When your session expires, the exception code INVALID_SESSION_ID is returned. If this happens, you must invoke the [login\(\)](#) call again.

More About Error Handling

For more information about errors, see the following topics:

- [API Fault Element](#)
- [ExceptionCode](#)
- [Error](#)

CHAPTER 5 Security and the API

In this chapter ...

- [User Authentication](#)
- [User Profile and Permission Sets Configuration](#)
- [Security Token](#)
- [Sharing](#)
- [Implicit Restrictions for Objects and Fields](#)
- [API Access in Force.com AppExchange Packages](#)
- [Outbound Port Restrictions](#)

Client applications that access your organization's Salesforce data are subject to the same security protections that are used in the Salesforce user interface. Additional protection is available for organizations that install Force.com AppExchange managed packages if those packages contain components that access Salesforce via the API.

User Authentication

Client applications must log in using valid credentials for an organization. The server authenticates these credentials and, if valid, provides the client application with:

- a `sessionId` that must be set into the session header so that all subsequent calls to the Web service are authenticated
- a URL address (`serverUrl`) for the client application's Web service requests

Salesforce supports only the Transport Layer Security (TLS) protocol and frontdoor.jsp. Ciphers must have a key length of at least 128 bits.

User Profile and Permission Sets Configuration

An organization's Salesforce administrator controls the availability of various features and views by configuring profiles and permission sets, and assigning users to them. To access the API (to issue calls and receive the call results), a user must be granted the "API Enabled" permission. Client applications can query or update only those objects and fields to which they have access via the permissions of the logged-in user.

To create, edit, or delete a profile, from Setup, enter *Profiles* in the Quick Find box, then select **Profiles** in the Salesforce user interface. To create, edit, or delete a permission set, from Setup, enter *Permission Sets* in the Quick Find box, then select **Permission Sets**.



Note: The Web services WSDL files return all available objects and fields for an organization.

Security Token

When users log in to Salesforce via the user interface, the API, or a desktop client such as Salesforce for Outlook, Connect Offline, Connect for Office, or the Data Loader, Salesforce confirms that the login is authorized as follows.

1. Salesforce checks whether the user's profile has login hour restrictions. If login hour restrictions are specified for the user's profile, any login outside the specified hours is denied.
2. If the user has the "Two-Factor Authentication for User Interface Logins" permission, Salesforce prompts the user for a second form of authentication upon logging in. If the user's account isn't already connected to a mobile authenticator app such as Salesforce Authenticator, Salesforce first prompts the user to connect the app.
3. If the user has the "Two-Factor Authentication for API Logins" permission and has connected an authenticator app to the account, Salesforce returns an error if the user uses the standard security token. The user has to enter a verification code (time-based one-time password) generated by the authenticator app instead.
4. Salesforce then checks whether the user's profile has IP address restrictions. If IP address restrictions are defined for the user's profile, logins from an undesignated IP address are denied, and logins from a specified IP address are allowed. If the **Enforce login IP ranges on every request** session setting is enabled, the IP address restrictions are enforced for each page request, including requests from client applications.
5. If profile-based IP address restrictions are not set, Salesforce checks whether the user is logging in from a device used to access Salesforce before.
 - If the user's login is from a device and browser that Salesforce recognizes, the login is allowed.
 - If the user's login is from an IP address in your organization's trusted IP address list, the login is allowed.
 - If the user's login is not from a trusted IP address or a device and browser Salesforce recognizes, the login is blocked.

Whenever a login is blocked or returns an API login fault, Salesforce has to verify the user's identity:

- For access via the user interface, the user is prompted to verify using Salesforce Authenticator (version 2 or later), or to enter a verification code.



Note: Users aren't asked for a verification code the first time they log in to Salesforce.

- For access via the API or a client, users must add their security token to the end of their password to log in. Or, if "Two-Factor Authentication on API Logins" is set on the user profile, users enter a verification code generated by an authenticator app.

A security token is an automatically generated key from Salesforce. For example, if a user's password is *mypassword*, and the security token is *XXXXXXXXXX*, the user must enter *mypasswordXXXXXXXXXX* to log in. Or some client applications have a separate field for the security token.

Users can obtain their security token by changing their password or resetting their security token via the Salesforce user interface. When a user changes a password or resets a security token, Salesforce sends a new security token to the email address on the user's Salesforce record. The security token is valid until the user resets the security token, changes a password, or has a password reset.



Tip: Before you access Salesforce from a new IP address, we recommend that you get your security token from a trusted network using **Reset My Security Token**.

For more information about tokens, see "Reset Your Security Token" in the Salesforce online help.

When a user's password is changed, the user's security token is automatically reset. The user will experience a blocked login until he or she adds the automatically-generated security token to the end of his or her password or enters the new password after the administrator adds their IP address to the organization's list of trusted IP addresses.

If Single Sign-On (SSO) is enabled for your organization, users who access the API or a desktop client cannot log in to Salesforce unless their IP address is included on your organization's list of trusted IP addresses or on their profile, if their profile has IP address restrictions set. Furthermore, the delegated authentication authority usually handles login lockout policies for users with the "Uses Single Sign-On" permission. However, if the security token is enabled for your organization, then your organization's login lockout settings determine the number of times a user can attempt to log in with an invalid security token before being locked out of Salesforce. For more information, see "Setting Login Restrictions" and "Setting Password Policies" in the online help.

Sharing

In the Salesforce user interface, sharing refers to the act of granting read or write access to a user or group so that they can view or edit a record owned by other users, if the default organization access levels do not otherwise permit such access. All API calls respect the sharing model.

The following table describes the types of access levels.

API Value	Salesforce User Interface Label	API Picklist Label	Description
None	Private	Private	Only the record owner and Users above that role in the hierarchy can view and edit the record.
Read	Read Only	Read Only	All Users and Groups can view the record but not edit it. Only the owner and users above that role in the hierarchy can edit the record.
Edit	Read/Write	Read/Write	All Users and Groups can view and edit the record.

API Value	Salesforce User Interface Label	API Picklist Label	Description
ReadEditTransfer	Read/Write/Transfer	Read/Write/Transfer	All Users and Groups can view, edit, delete, and transfer the record. (Only available for cases and leads as an organization-wide default setting.)
All	Full Access	Owner	All Users and Groups can view, edit, transfer, delete, and share the record. (Only available for campaigns as an organization-wide default setting.)
ControlledByParent	Controlled by Parent	Controlled By Parent	(Contacts only.) All Users and Groups can perform an action (such as view, edit, or delete) on the contact based on whether he or she can perform that same action on the record associated with it.

Not all access levels are available for every object. See the Fields table for each object to learn which access levels are available, as well as other sharing details specific to that object.

For more information about sharing in general, see the Salesforce online help.

 **Note:** In the API, you can create and update objects such as [AccountShare](#) and [OpportunityShare](#) that define sharing entries for records.

Implicit Restrictions for Objects and Fields

Certain objects can be created or deleted only in the Salesforce user interface. Other objects are read-only—client applications cannot `create()`, `delete()`, or `update()` such objects. Similarly, certain fields within some objects can be specified on `create()` but not on `update()`. Other fields are read-only—client applications cannot specify field values in `create()` or `update()` calls. For more information, see the respective object descriptions in [Object Basics](#).

API Access in Force.com AppExchange Packages

The API allows access to objects and calls based on the permissions of the user who logs into the API. To prevent security issues from arising when installed packages have components that access data via the API, Salesforce provides additional security:

- When a developer creates an AppExchange package with components that access the API, the developer can restrict the API access for those components.
- When an administrator installs an AppExchange package, the administrator can accept or reject the access. Rejecting the access cancels the installation.
- After an administrator installs a package, the administrator can restrict the API access of components in the package that access the API.

Editing API access for a package is done in the Salesforce user interface. For more information, see “Manage API and Dynamic Apex Access in Packages” in the Salesforce online help.

API access for a package affects the API requests originating from components within the package; it determines the objects that the API requests can access. If the API access for a package is not defined, then the objects that the API requests have access to are determined by the user's permissions.

The API access for a package never allows users to do more than the permissions granted to the user. API access in a package only reduces what the user's permissions allow.

Choosing **Restricted** for the **API Access** setting in a package affects the following:

- API access in a package overrides the following user permissions:
 - Author Apex
 - Customize Application
 - Edit HTML Templates
 - Edit Read Only Fields
 - Manage Billing
 - Manage Call Centers
 - Manage Categories
 - Manage Custom Report Types
 - Manage Dashboards
 - Manage Letterheads
 - Manage Package Licenses
 - Manage Public Documents
 - Manage Public List Views
 - Manage Public Reports
 - Manage Public Templates
 - Manage Users
 - Transfer Record
 - Use Team Reassignment Wizards
 - View Setup and Configuration
 - Weekly Export Data
- If **Read**, **Create**, **Edit**, and **Delete** access are not selected in the API access setting for objects, users do not have access to those objects from the package components, even if the user has the "Modify All Data" and "View All Data" permissions.
- A package with **Restricted** API access can't create new users.
- Salesforce denies access to Web service and `executeanonymous` requests from an AppExchange package that has **Restricted** access.

The following considerations also apply to API access in packages:

- Workflow rules and Apex triggers fire regardless of API access in a package.
- If a component is in more than one package in an organization, API access is unrestricted for that component in all packages in the organization regardless of the access setting.
- If Salesforce introduces a new standard object after you select restricted access for a package, access to the new standard object is not granted by default. You must modify the restricted access setting to include the new standard object.
- When you upgrade a package, changes to the API access are ignored even if the developer specified them. This ensures that the administrator installing the upgrade has full control. Installers should carefully examine the changes in package access in each upgrade during installation and note all acceptable changes. Then, because those changes are ignored, the administrator should manually apply any acceptable changes after installing an upgrade.

- S-controls are served by Salesforce and rendered inline in Salesforce. Because of this tight integration, there are several means by which an s-control in an installed package could escalate its privileges to the user's full privileges. In order to protect the security of organizations that install packages, s-controls have the following limitations:
 - For packages you are developing (that is, not installed from AppExchange), you can only add s-controls to packages with the default `Unrestricted` API access. Once a package has an s-control, you cannot enable `Restricted` API access.
 - For packages you have installed, you can enable access restrictions even if the package contains s-controls. However, access restrictions provide only limited protection for s-controls. Salesforce recommends that you understand the JavaScript in an s-control before relying on access restriction for s-control security.
 - If an installed package has `Restricted` API access, upgrades will be successful only if the upgraded version does not contain any s-controls. If s-controls are present in the upgraded version, you must change the currently installed package to `Unrestricted` API access.

To manage API access to packages, see "Manage API and Dynamic Apex Access in Packages" in the Salesforce online help.



Note: XML-RPC requests that originate from restricted packages will be denied access.

Outbound Port Restrictions

For security reasons, Salesforce restricts the outbound ports you may specify to one of the following:

- 80: This port only accepts HTTP connections.
- 443: This port only accepts HTTPS connections.
- 1024–66535 (inclusive): These ports accept HTTP or HTTPS connections.

The port restriction applies to any feature where a port is specified, for example outbound messages, AJAX proxy, or single-sign on.

CHAPTER 6 Using the Partner WSDL

In this chapter ...

- [Obtaining the Partner WSDL File](#)
- [Calls and the Partner WSDL](#)
- [Objects, Fields, and Field Data and the Partner WSDL](#)
- [Queries and the Partner WSDL](#)
- [Namespaces in the Partner WSDL](#)
- [Package Versions and the Partner WSDL](#)
- [User Interface Themes](#)
- [Examples Using the Partner WSDL](#)

The API provides two WSDLs to choose from:

- **Enterprise Web Services WSDL**—Used by enterprise developers to build client applications for a single Salesforce organization. The enterprise WSDL is strongly typed, which means that it contains objects and fields with specific data types, such as `int` and `string`. Customers who use the enterprise WSDL document must download and re-consume it when changes are made to the custom objects or fields in their org or when they want to use a different version of the API. To access the current WSDL for your organization, log in to your Salesforce organization and from Setup, enter `Generate Enterprise WSDL` in the Quick Find box, then select **Generate Enterprise WSDL**.
- **Partner Web Services WSDL**—Used for client applications that are metadata-driven and dynamic in nature. It is particularly—but not exclusively—useful to Salesforce partners who are building client applications for multiple organizations. As a loosely typed representation of the Salesforce data model that works with name-value pairs of field names and values instead of specific data types, it can be used to access data within any organization. This WSDL is most appropriate for developers of clients that can issue a query call to get information about an object before the client acts on the object. The partner WSDL document needs to be downloaded and consumed only once per version of the API. To access the current WSDL for your organization, log in to your Salesforce organization and from Setup, enter `Generate Partner WSDL` in the Quick Find box, then select **Generate Partner WSDL**.

In general, the enterprise WSDL is more straightforward to use, while the partner WSDL is more flexible and dynamically adaptable to different organizations, allowing you to write a single application that can be used for multiple users and multiple organizations.

Obtaining the Partner WSDL File

To use the partner WSDL, download a copy of the file using either of the following methods:

- Obtain it from your organization's Salesforce administrator, or
- Generate from Setup in Salesforce (enter **API** in the **Quick Find** box, then select **API**) according to the instructions in [Step 2: Generate or Obtain the Web Service WSDL](#).

While the enterprise WSDL file needs to be regenerated whenever custom fields or custom objects are added to an organization's Salesforce information, the partner WSDL file remains the same regardless of underlying changes in the organization's Salesforce data.

Calls and the Partner WSDL

The partner WSDL file defines exactly the same API calls found in the enterprise WSDL file. A client application using the partner WSDL will likely use the following API calls to determine an organization's metadata:

Task / Call	Description
<code>describeGlobal()</code>	Retrieves a list of available objects for your organization's data.
<code>describeLayout()</code>	Retrieves metadata about page layouts for the specified object type.
<code>describeSObject()</code>	<code>describeSObject()</code> has been superseded by <code>describeSObjects()</code> .
<code>describeSObjects()</code>	Use to obtain metadata for a given object. You can first call to retrieve a list of all objects for your organization, then iterate through the list and use to obtain metadata about individual objects.
<code>describeTabs()</code>	In the user interface, users have access to standard apps (and may also have access to custom apps) as listed in the Force.com app menu at the top of the page. Selecting a standard app or custom app in the user interface allows the user to switch between the listed apps at any time.

To explore an organization's metadata, a client application can:

1. Call `describeGlobal()` to obtain a list of available objects.
2. In the returned `DescribeGlobalResult` object, retrieve an array of `DescribeGlobalSObjectResult` objects by calling `sobjects`.
3. Get the sObject type name for each returned sObject by calling `name` on the `DescribeGlobalSObjectResult` objects.
4. The `DescribeGlobalSObjectResult` object provides some metadata about the sObject, such as whether the sObject is createable or updateable. If you want to get more information about particular sObjects, like their fields and child relationships, call `describeSObjects()` by passing it an array of the sObject type names that you're interested in obtaining more information about.

sObject Reference Reuse

An sObject reference can't be reused within a single operation.

Use a different reference. For example, the following code snippet creates an account and contact with a custom field and an event using two different references:

```
SObject account = new com.sforce.soap.partner.sobject.wsc.SObject();
account.setType("Account");
```

```

account.setField("Name", "myAccount");
account.setField("XID1__c", "1");
SObject refAcc1 = new com.sforce.soap.partner.sobject.wsc.SObject();
refAcc1.setType("Account");
refAcc1.setField("XID1__c", "1");
SObject refAcc2 = new com.sforce.soap.partner.sobject.wsc.SObject();
refAcc2.setType("Account");
refAcc2.setField("XID1__c", "1");

SObject contact = new com.sforce.soap.partner.sobject.wsc.SObject();
contact.setType("Contact");
contact.setField("LastName", "LName");
contact.setField("XID2__c", "2");
contact.setField("Account", refAcc1);
SObject refCon = new com.sforce.soap.partner.sobject.wsc.SObject();
contact.setType("Contact");
contact.setField("XID2__c", "2");

SObject event = new com.sforce.soap.partner.sobject.wsc.SObject();
contact.setType("Event");
contact.setField("Subject", "myEvent");
contact.setField("ActivityDateTime", Calendar.getInstance());
contact.setField("DurationInMinutes", 60);
contact.setField("Who", refCon);
contact.setField("What", refAcc2);

client.create(new SObject[] { account, contact, event}); // exception thrown here

```

Any call that takes a parameter of the form `sObject[] sObjects` is subject to this limitation.

Objects, Fields, and Field Data and the Partner WSDL

The enterprise WSDL file defines all the specific objects (such as Account and Contact) in a Salesforce org. In contrast, the partner WSDL file defines a single, generic object (`sObject`) that represents all the objects. For a particular object, its type is defined in the `name` field in the returned `DescribeSObjectResult`.

With the partner WSDL, your client application code handles fields as arrays of name-value pairs that represent the field data. When referring to the name of an individual field, use the value in its `name` field of the `Field` type in the `DescribeSObjectResult`.

Languages vary in the way they handle name-value pairs and map typed values to the primitive XML data types defined in SOAP messages. With the enterprise WSDL, the mapping is handled implicitly. With the partner WSDL, however, you manually manage values and data types when building client applications. Specify the object type before you assign field values. When specifying the value of a particular field, use a value that is valid for the field (range, format, and data type). Make sure that you understand the mapping between data types in your programming language and XML primitive data types. See [SOAPType](#) for more information.

Queries and the Partner WSDL

When using the `query()` call with the partner WSDL, consider the following guidelines:

- The `queryString` parameter is case-insensitive. The API will accept field names in the `fieldList` using any combination of uppercase and lowercase letters. However, in the `QueryResult`, the case of field names (both predefined and custom fields) will match exactly.

the value in the `name` field of the `Field` type in the `DescribeSObjectResult`. It is recommended that you use the proper case when specifying fields in the `fieldList`.

- For the partner WSDL, the ordering of fields in the `QueryResult` is determined by the field order in the `fieldList`, not the field order in the WSDL file.
- The `fieldList` cannot contain duplicate field names. For example:
 - Invalid (returns an error): "SELECT Firstname, Lastname, Firstname FROM User"
 - Valid: "SELECT Firstname, Lastname FROM User"
- The `QueryResult` always contains all of the fields specified in the `fieldList`, even if some of the fields contain no data (`null`). Although SOAP allows you to omit fields that contain no values in the result set, the API always returns an array containing all fields.
- If you use the partner WSDL, a query that includes ID will return the ID field twice in the SOAP XML response data. Similarly, a query that does not include ID will return a single null ID field in the SOAP XML response data. For example, a query for `SELECT ID, FirstName, LastName FROM Contact` might return a SOAP XML response with records like:

```
<records xsi:type="sf:sObject" xmlns="urn:partner.soap.sforce.com">
  <sf:type>Contact</sf:type>
  <sf:Id>0038000000FrjoBQRW</sf:Id>
  <sf:Id>0038000000FrjoBQRW</sf:Id>
  <sf:FirstName>John</sf:FirstName>
  <sf:LastName>Smith</sf:LastName>
</records>
```

This is expected behavior and something to be aware of if you are accessing the full SOAP XML response data and not using WSC to access the web service response.

Namespaces in the Partner WSDL

In XML, every tag has a defined namespace. In the `enterprise.wsdl`, namespaces are handled implicitly. When using API calls with the partner WSDL, however, you need to explicitly specify the correct namespaces for API calls, objects, and fields, and faults. This rule applies to predefined and custom objects and fields.

For	Namespace
API Calls	urn:partner.soap.sforce.com
sObjects	urn:sobject.partner.soap.sforce.com
Fields	urn:sobject.partner.soap.sforce.com
Faults	urn:fault.partner.soap.sforce.com

Package Versions and the Partner WSDL

The partner WSDL is loosely typed. This makes it more flexible for partners who want to integrate with multiple organizations. Default package versions for API calls provide fallback settings if package versions are not provided by an API call.

The behavior of a package in partner API calls is defined by the latest installed package version if the default value (`Not Specified`) is selected for the installed package. This means that behavior of package components, such as an Apex trigger, could change when a package is upgraded and that change would immediately impact the integration. Subscribers may want to select a specific version for

an installed package for all partner API calls from client applications to ensure that subsequent installations of package versions do not affect their existing integrations.

An API client developer should communicate with the administrator of the default partner package version settings if these are two different roles in your organization and the developer recommends changing the settings. Alternatively, an API client developer can set the package versions in the [PackageVersionHeader](#) SOAP header for the client.

A partner that is developing a package that references another package should always supply version information for the base package in their partner API calls. This ensures that the extension package is not affected by a component being deprecated in the base package.

The package version information for partner API calls is determined by the first match in the following settings.

1. The PackageVersionHeader SOAP header.
2. An API call from a Visualforce page uses the package versions set for the Visualforce page.
3. The default partner package version settings.

To configure default package versions for API calls with the partner WSDL, see [Package Version Settings](#).

User Interface Themes

Back in the Winter '06 release, Salesforce started supporting multiple user interface themes, allowing you to use different sets of icons and colors for the user interface. But these user interface themes do not apply when your org is using Lightning Experience.

Two user interface themes match the earlier iterations of Salesforce.

- Theme3—The “Salesforce Classic 2010 user interface theme.” This interface was previously referred to as “Salesforce” or “new user interface theme.” You might also be familiar with it as the Salesforce *Aloha* interface.
- Theme2—The “Salesforce Classic 2005 user interface theme.” This interface was previously referred to as “Salesforce Classic” or the “classic user interface theme.”

The `getUserInfo()` call returns a `getUserInfoResult` object, which includes the `userUiSkin` property. This property informs you of the user’s current user interface theme.

Use the `describeQuickActions()`, `describeTabs()`, and `describeTheme()` calls and their return types to get information on theme icons and colors.

Style sheets are available to mimic the look and feel of the older user interfaces. For more information, see Styling Visualforce Pages in the *Visualforce Developer’s Guide*. But if you’re planning to switch to Lightning Experience, consider the Lightning Component framework, our new UI framework. See the “[Lightning Components](#)” module in the [Developer Trail - Lightning Experience Trailhead](#) trail to learn more.

Editions

Available in: Salesforce Classic and earlier

Examples Using the Partner WSDL

This section includes examples in Java and C# for making API calls using the partner WSDL. Before running these samples, perform the following steps in the quick start tutorial to get the partner WSDL file and generate the proxy client code for your development environment.

- [Step 2: Generate or Obtain the Web Service WSDL](#)
- [Step 3: Import the WSDL File Into Your Development Platform](#)

After you generate the proxy client code and set up your development environment, you can start writing your client application. First, your application needs to log into the Salesforce service using the partner authentication endpoint. After a successful login, you can execute the sample methods.

For your convenience, template classes are provided, one in Java and one in C#, that make a login call. You can use them to execute the sample methods provided later in this section.

Sample template class for Java: This sample prompts the user to enter the username, password, and authentication endpoint. Next, it logs the user in. For the authentication endpoint URL, pass in the endpoint found in the partner WSDL file.

```
import com.sforce.soap.partner.PartnerConnection;
import com.sforce.soap.partner.sobject.*;
import com.sforce.soap.partner.*;
import com.sforce.ws.ConnectorConfig;
import com.sforce.ws.ConnectionException;
import com.sforce.soap.partner.Error;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.BufferedReader;
import java.util.*;

public class PartnerSamples {
    PartnerConnection partnerConnection = null;
    private static BufferedReader reader =
        new BufferedReader(new InputStreamReader(System.in));

    public static void main(String[] args) {
        PartnerSamples samples = new PartnerSamples();
        if (samples.login()) {
            // Add calls to the methods in this class.
            // For example:
            // samples.querySample();
        }
    }

    private String getUserInput(String prompt) {
        String result = "";
        try {
            System.out.print(prompt);
            result = reader.readLine();
        } catch (IOException ioe) {
            ioe.printStackTrace();
        }
        return result;
    }

    private boolean login() {
        boolean success = false;
        String username = getUserInput("Enter username: ");
        String password = getUserInput("Enter password: ");
        String authEndPoint = getUserInput("Enter auth end point: ");

        try {
            ConnectorConfig config = new ConnectorConfig();
            config.setUsername(username);
            config.setPassword(password);

            config.setAuthEndpoint(authEndPoint);
        }
    }
}
```

```

        config.setTraceFile("traceLogs.txt");
        config.setTraceMessage(true);
        config.setPrettyPrintXml(true);

        partnerConnection = new PartnerConnection(config);

        success = true;
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    } catch (FileNotFoundException fnfe) {
        fnfe.printStackTrace();
    }

    return success;
}

// Add your methods here.
//
}

```

Sample template class for C#: This sample prompts the user to enter the username and password. Next, it logs the user in. The project name for this sample is assumed to be *TemplatePartner* and the Web reference name *sforce*. If these values are different for your project, make sure to change the using directive to appropriate values for your project: `using your_project_name.web_reference_name;`.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web.Services.Protocols;
using System.Collections;
using TemplatePartner.sforce;

namespace TemplatePartner
{
    class PartnerSamples
    {
        private SforceService binding;

        static void Main(string[] args)
        {
            PartnerSamples samples = new PartnerSamples();
            if (samples.login())
            {
                // Add calls to the methods in this class.
                // For example:
                // samples.querySample();
            }
        }

        private bool login()
        {
            Console.Write("Enter username: ");

```

```
string username = Console.ReadLine();
Console.Write("Enter password: ");
string password = Console.ReadLine();

// Create a service object
binding = new SforceService();

// Timeout after a minute
binding.Timeout = 60000;

// Try logging in
LoginResult lr;
try
{
    Console.WriteLine("\nLogging in...\n");
    lr = binding.login(username, password);
}

// ApiFault is a proxy stub generated from the WSDL contract when
// the web service was imported
catch (SoapException e)
{
    // Write the fault code to the console
    Console.WriteLine(e.Code);

    // Write the fault message to the console
    Console.WriteLine("An unexpected error has occurred: " + e.Message);

    // Write the stack trace to the console
    Console.WriteLine(e.StackTrace);

    // Return False to indicate that the login was not successful
    return false;
}

// Check if the password has expired
if (lr.passwordExpired)
{
    Console.WriteLine("An error has occurred. Your password has expired.");
    return false;
}

// Set the returned service endpoint URL
binding.Url = lr.serverUrl;

// Set the SOAP header with the session ID returned by
// the login result. This will be included in all
// API calls.
binding.SessionHeaderValue = new SessionHeader();
binding.SessionHeaderValue.sessionId = lr.sessionId;

// Return true to indicate that we are logged in, pointed
// at the right URL and have our security token in place.
```

```

        return true;
    }

    //
    // Add your methods here.
    //
}
```

This partner WSDL samples are:

- [Sample query and queryMore Calls](#)
- [Sample search Call](#)
- [Sample create Call](#)
- [Sample update Call](#)

Sample query and queryMore Calls

The following Java and C# examples show usage of the `query()` and `queryMore()` calls for the partner WSDL. Each example sets the batch size of the query to 250 items returned. It then performs a query call to get the first name and last name of all contacts and iterates through the contact records returned. For each contact, it writes the contact's first name and last name to the output, or only the last name if the first name is null. Finally, if there are more items to be returned by the query, as indicated by a `QueryResult.done` property value of `false`, it calls `queryMore()` to get the next batch of items, and repeats the process until no more records are returned.

To execute the sample method, you can use the corresponding Java or C# template class provided in [Examples Using the Partner WSDL](#).

Java Example

```

public void querySample() {
    try {
        // Set query batch size
        partnerConnection.setQueryOptions(250);

        // SOQL query to use
        String soqlQuery = "SELECT FirstName, LastName FROM Contact";
        // Make the query call and get the query results
        QueryResult qr = partnerConnection.query(soqlQuery);

        boolean done = false;
        int loopCount = 0;
        // Loop through the batches of returned results
        while (!done) {
            System.out.println("Records in results set " + loopCount++
                + " - ");
            SObject[] records = qr.getRecords();
            // Process the query results
            for (int i = 0; i < records.length; i++) {
                SObject contact = records[i];
                Object firstName = contact.getField("FirstName");
                Object lastName = contact.getField("LastName");
                if (firstName == null) {
                    System.out.println("Contact " + (i + 1) +
                        " has no first name");
                } else {
                    System.out.println("Contact " + (i + 1) +
                        " has first name " + firstName);
                }
            }
            if (qr.isLastBatch()) {
                done = true;
            } else {
                qr = partnerConnection.queryMore();
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

        ": " + lastName
    );
} else {
    System.out.println("Contact " + (i + 1) + ":" +
        firstName + " " + lastName);
}
}
if (qr.isDone()) {
    done = true;
} else {
    qr = partnerConnection.queryMore(qr.getQueryLocator());
}
}
} catch(ConnectionException ce) {
    ce.printStackTrace();
}
System.out.println("\nQuery execution completed.");
}

```

C# Example

```

public void querySample()
{
    try
    {
        QueryResult qr = null;
        binding.QueryOptionsValue = new sforce.QueryOptions();
        binding.QueryOptionsValue.batchSize = 250;
        binding.QueryOptionsValue.batchSizeSpecified = true;

        qr = binding.query("SELECT FirstName, LastName FROM Contact");

        bool done = false;
        int loopCount = 0;
        while (!done)
        {
            Console.WriteLine("\nRecords in results set " +
                Convert.ToString(loopCount++) +
                " - ");
            // Process the query results
            for (int i = 0; i < qr.records.Length; i++)
            {
                sforce.sObject con = qr.records[i];
                string fName = con.Any[0].InnerText;
                string lName = con.Any[1].InnerText;
                if (fName == null)
                    Console.WriteLine("Contact " + (i + 1) + ":" + lName);
                else
                    Console.WriteLine("Contact " + (i + 1) + ":" + fName +
                        " " + lName);
            }
        }

        if (qr.done)
    }
}

```

```

        done = true;
    else
        qr = binding.queryMore(qr.queryLocator);
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " + e.Message +
        " Stack trace: " + e.StackTrace);
}
Console.WriteLine("\nQuery execution completed.");
}

```

Sample search Call

The following Java and C# examples show how to use the [search\(\)](#) call for the partner WSDL. Each example accepts a phone number string value that is used in the SOQL query. The search call looks for phone fields that match the passed in phone value in all contacts, leads, and accounts. Next, the example iterates through the returned search results that contain the matching records, adds them to arrays, and writes their field values to the console. The record fields returned correspond to the fields specified in the SOQL query for each record type.

To execute the sample method, you can use the corresponding Java or C# template class provided in [Examples Using the Partner WSDL](#).

Java Example

```

public void searchSample(String phoneNumber) {
    try {
        // Example of phoneNumber format: 4155551212
        String soslQuery =
            "FIND {" + phoneNumber + "} IN Phone FIELDS " +
            "RETURNING " +
            "Contact(Id, Phone, FirstName, LastName), " +
            "Lead(Id, Phone, FirstName, LastName), " +
            "Account(Id, Phone, Name)";
        // Perform SOSL query
        SearchResult sResult = partnerConnection.search(soslQuery);
        // Get the records returned by the search result
        SearchRecord[] records = sResult.getSearchRecords();
        // Create lists of objects to hold search result records
        List<SObject> contacts = new ArrayList<SObject>();
        List<SObject> leads = new ArrayList<SObject>();
        List<SObject> accounts = new ArrayList<SObject>();

        // Iterate through the search result records
        // and store the records in their corresponding lists
        // based on record type.
        if (records != null && records.length > 0) {
            for (int i = 0; i < records.length; i++) {
                SObject record = records[i].getRecord();
                if (record.getType().toLowerCase().equals("contact")) {
                    contacts.add(record);
                } else if (record.getType().toLowerCase().equals("lead")) {

```

```

        leads.add(record);
    } else if (record.getType().toLowerCase().equals("account")) {
        accounts.add(record);
    }
}
// Display the contacts that the search returned
if (contacts.size() > 0) {
    System.out.println("Found " + contacts.size() +
        " contact(s):");
    for (SObject contact : contacts) {
        System.out.println(contact.getId() + " - " +
            contact.getField("FirstName") + " " +
            contact.getField("LastName") + " - " +
            contact.getField("Phone")
        );
    }
}
// Display the leads that the search returned
if (leads.size() > 0) {
    System.out.println("Found " + leads.size() +
        " lead(s):");
    for (SObject lead : leads) {
        System.out.println(lead.getId() + " - " +
            lead.getField("FirstName") + " " +
            lead.getField("LastName") + " - " +
            lead.getField("Phone")
        );
    }
}
// Display the accounts that the search returned
if (accounts.size() > 0) {
    System.out.println("Found " +
        accounts.size() + " account(s):");
    for (SObject account : accounts) {
        System.out.println(account.getId() + " - " +
            account.getField("Name") + " - " +
            account.getField("Phone")
        );
    }
} else {
    // The search returned no records
    System.out.println("No records were found for the search.");
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

C# Example

```

public void searchSample(String phoneNumber)
{

```

```

try
{
    // Example of phoneNumber format: 4155551212
    String soslQuery =
        "FIND {" + phoneNumber + "} IN Phone FIELDS " +
        "RETURNING " +
        "Contact(Id, Phone, FirstName, LastName), " +
        "Lead(Id, Phone, FirstName, LastName), " +
        "Account(Id, Phone, Name)";

    // Perform SOSL query
    SearchResult sResult = binding.search(soslQuery);
    // Get the records returned by the search result
    SearchRecord[] records = sResult.searchRecords;
    // Create lists of objects to hold search result records
    ArrayList contacts = new System.Collections.ArrayList();
    ArrayList leads = new System.Collections.ArrayList();
    ArrayList accounts = new System.Collections.ArrayList();

    // Iterate through the search result records
    // and store the records in their corresponding lists
    // based on record type.
    if ((records != null) && (records.Length > 0))
    {
        for (int i = 0; i < records.Length; i++)
        {
            sObject record = records[i].record;

            if (record.type.ToLower().Equals("contact"))
            {
                contacts.Add(record);
            }
            else if (record.type.ToLower().Equals("lead"))
            {
                leads.Add(record);
            }
            else if (record.type.ToLower().Equals("account"))
            {
                accounts.Add(record);
            }
        }
        // Display the contacts that the search returned
        if (contacts.Count > 0)
        {
            Console.WriteLine("Found " + contacts.Count + " contact(s):");
            for (int i = 0; i < contacts.Count; i++)
            {
                sObject c = (sObject)contacts[i];
                Console.WriteLine(c.Any[0].InnerText + " - " +
                    c.Any[2].InnerText + " " +
                    c.Any[3].InnerText + " - " + c.Any[1].InnerText);
            }
        }
        // Display the leads that the search returned
        if (leads.Count > 0)
    }
}

```

```

        {
            Console.WriteLine("Found " + leads.Count + " lead(s):");
            for (int i = 0; i < leads.Count; i++)
            {
                sObject l = (sObject)leads[i];
                Console.WriteLine(l.Any[0].InnerText + " - " +
                    l.Any[2].InnerText + " " +
                    l.Any[3].InnerText + " - " + l.Any[1].InnerText);
            }
        }
        // Display the accounts that the search returned
        if (accounts.Count > 0)
        {
            Console.WriteLine("Found " + accounts.Count + " account(s):");
            for (int i = 0; i < accounts.Count; i++)
            {
                sObject a = (sObject)accounts[i];
                Console.WriteLine(a.Any[0].InnerText + " - " +
                    a.Any[2].InnerText + " - " +
                    a.Any[1].InnerText);
            }
        }
    }
    else
    {
        // The search returned no records
        Console.WriteLine("No records were found for the search.");
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " + e.Message +
        " Stack trace: " + e.StackTrace);
}
}

```

Sample create Call

The following Java and C# examples show how to use the `create()` call for the partner WSDL. Each example creates a contact record with several fields. It iterates through the results of the create call and checks whether the operation was successful or not. If the create operation was successful, it writes the ID of the contact created to the console. Otherwise, it iterates through the errors and writes details of each error to the console. In this case, the output of the example is the ID of the new contact.

To execute the sample method, you can use the corresponding Java or C# template class provided in [Examples Using the Partner WSDL](#).

Java Example

```
public String createSample() {  
    String result = null;  
    try {  
        // Create a new sObject of type Contact  
        // and fill out its fields.  
        SObject contact = new SObject();
```

```

contact.setType("Contact");
contact.setField("FirstName", "Otto");
contact.setField("LastName", "Jespersen");
contact.setField("Salutation", "Professor");
contact.setField("Phone", "(999) 555-1234");
contact.setField("Title", "Philologist");

// Add this sObject to an array
SObject[] contacts = new SObject[1];
contacts[0] = contact;
// Make a create call and pass it the array of sObjects
SaveResult[] results = partnerConnection.create(contacts);

// Iterate through the results list
// and write the ID of the new sObject
// or the errors if the object creation failed.
// In this case, we only have one result
// since we created one contact.
for (int j = 0; j < results.length; j++) {
    if (results[j].isSuccess()) {
        result = results[j].getId();
        System.out.println(
            "\nA contact was created with an ID of: " + result
        );
    } else {
        // There were errors during the create call,
        // go through the errors array and write
        // them to the console
        for (int i = 0; i < results[j].getErrors().length; i++) {
            Error err = results[j].getErrors()[i];
            System.out.println("Errors were found on item " + j);
            System.out.println("Error code: " +
                err.getStatusCode().toString());
            System.out.println("Error message: " + err.getMessage());
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
return result;
}

```

C# Example

```

public void createSample()
{
    try
    {
        // Create a new sObject of type Contact
        // and fill out its fields.
        sObject contact = new sforce.sObject();
        System.Xml.XmlElement[] contactFields = new System.Xml.XmlElement[6];

```

```
// Create the contact's fields
System.Xml.XmlDocument doc = new System.Xml.XmlDocument();
contactFields[0] = doc.CreateElement("FirstName");
contactFields[0].InnerText = "Otto";
contactFields[1] = doc.CreateElement("LastName");
contactFields[1].InnerText = "Jespersen";
contactFields[2] = doc.CreateElement("Salutation");
contactFields[2].InnerText = "Professor";
contactFields[3] = doc.CreateElement("Phone");
contactFields[3].InnerText = "(999) 555-1234";
contactFields[4] = doc.CreateElement("Title");
contactFields[4].InnerText = "Philologist";

contact.type = "Contact";
contact.Any = contactFields;

// Add this sObject to an array
sObject[] contactList = new sObject[1];
contactList[0] = contact;

// Make a create call and pass it the array of sObjects
SaveResult[] results = binding.create(contactList);
// Iterate through the results list
// and write the ID of the new sObject
// or the errors if the object creation failed.
// In this case, we only have one result
// since we created one contact.
for (int j = 0; j < results.Length; j++)
{
    if (results[j].success)
    {
        Console.WriteLine("\nA contact was created with an ID of: "
            + results[j].id);
    }
    else
    {
        // There were errors during the create call,
        // go through the errors array and write
        // them to the console
        for (int i = 0; i < results[j].errors.Length; i++)
        {
            Error err = results[j].errors[i];
            Console.WriteLine("Errors were found on item " + j.ToString());
            Console.WriteLine("Error code is: " + err.statusCode.ToString());
            Console.WriteLine("Error message: " + err.message);
        }
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " + e.Message +
        " Stack trace: " + e.StackTrace);
```

```
}
```

Sample update Call

The following Java and C# examples show how to use the `update()` call for the Partner WSDL. Each example takes the ID of the contact to update as an argument. It creates two sObject records of type Contact—one to hold the valid passed in ID and the other has an invalid ID. Next, it sets a new phone number for the valid contact and `null` for the last name of the invalid contact. It then makes the update call and iterates through the results. For a successful update operation, it writes the ID of the contact that got updated. For a failed update operation, it writes the details of all returned errors to the console. In this case, the output is the ID of the contact that was successfully updated and an error for the invalid contact update.

To execute the sample method, you can use the corresponding Java or C# template class provided in [Examples Using the Partner WSDL](#).

Java Example

```
public void updateSample(String id) {
    try {
        // Create an sObject of type contact
        SObject updateContact = new SObject();
        updateContact.setType("Contact");

        // Set the ID of the contact to update
        updateContact.setId(id);
        // Set the Phone field with a new value
        updateContact.setField("Phone", "(415) 555-1212");

        // Create another contact that will cause an error
        // because it has an invalid ID.
        SObject errorContact = new SObject();
        errorContact.setType("Contact");
        // Set an invalid ID on purpose
        errorContact.setId("SLFKJLFKJ");
        // Set the value of LastName to null
        errorContact.setFieldsToNull(new String[] {"LastName"});

        // Make the update call by passing an array containing
        // the two objects.
        SaveResult[] saveResults = partnerConnection.update(
            new SObject[] {updateContact, errorContact}
        );
        // Iterate through the results and write the ID of
        // the updated contacts to the console, in this case one contact.
        // If the result is not successful, write the errors
        // to the console. In this case, one item failed to update.
        for (int j = 0; j < saveResults.length; j++) {
            System.out.println("\nItem: " + j);
            if (saveResults[j].isSuccess()) {
                System.out.println("Contact with an ID of " +
                    saveResults[j].getId() + " was updated.");
            }
            else {

```

```
// There were errors during the update call,
// go through the errors array and write
// them to the console.
for (int i = 0; i < saveResults[j].getErrors().length; i++) {
    Error err = saveResults[j].getErrors()[i];
    System.out.println("Errors were found on item " + j);
    System.out.println("Error code: " +
        err.getStatusCode().toString());
    System.out.println("Error message: " + err.getMessage());
}
}
}
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

For more information about `setFieldsToNull` (or its equivalent in client tools other than WSC), see [fieldsToNull](#) and [Resetting Values to null](#).

C# Example

```
public void updateSample(String id) {  
    try  
    {  
        // Create an sObject of type contact  
        sObject updateContact = new sObject();  
        updateContact.type = "Contact";  
  
        // Set the ID of the contact to update  
        updateContact.Id = id;  
        // Set the Phone field to a new value.  
        // The Phone field needs to be created as an XML element.  
        System.Xml.XmlDocument doc = new System.Xml.XmlDocument();  
        System.Xml.XmlElement phoneField = doc.CreateElement("Phone");  
        phoneField.InnerText = "(415) 555-1212";  
  
        // Add the Phone field to the contact  
        updateContact.Any = new System.Xml.XmlElement[] {phoneField};  
  
        // Create another contact that will cause an error  
        // because it has an invalid ID.  
        sObject errorContact = new sObject();  
        errorContact.type = "Contact";  
        // Set an invalid ID on purpose  
        errorContact.Id = "SLFKJLJK";  
        // Set the value of LastName to null  
        errorContact.fieldsToNull = new String[] { "LastName" };  
  
        // Make the update call by passing an array containing  
        // the two objects.  
        SaveResult[] saveResults = binding.update(  
            new sObject[] {updateContact, errorContact});  
        // Iterate through the results and write the ID of
```

```
// the updated contacts to the console, in this case one contact.  
// If the result is not successful, write the errors  
// to the console. In this case, one item failed to update.  
for (int j = 0; j < saveResults.Length; j++) {  
    Console.WriteLine("\nItem: " + j);  
    if (saveResults[j].success)  
    {  
        Console.WriteLine("Contact with an ID of " +  
            saveResults[j].id + " was updated.");  
    }  
    else  
    {  
        // There were errors during the update call,  
        // go through the errors array and write  
        // them to the console.  
        for (int i = 0; i < saveResults[j].errors.Length; i++) {  
            Error err = saveResults[j].errors[i];  
            Console.WriteLine("Errors were found on item " + j.ToString());  
            Console.WriteLine("Error code: " +  
                err.statusCode.ToString());  
            Console.WriteLine("Error message: " + err.message);  
        }  
    }  
}  
}  
  
catch (SoapException e)  
{  
    Console.WriteLine("An unexpected error has occurred: " + e.Message +  
        " Stack trace: " + e.StackTrace);  
}  
}
```

REFERENCE

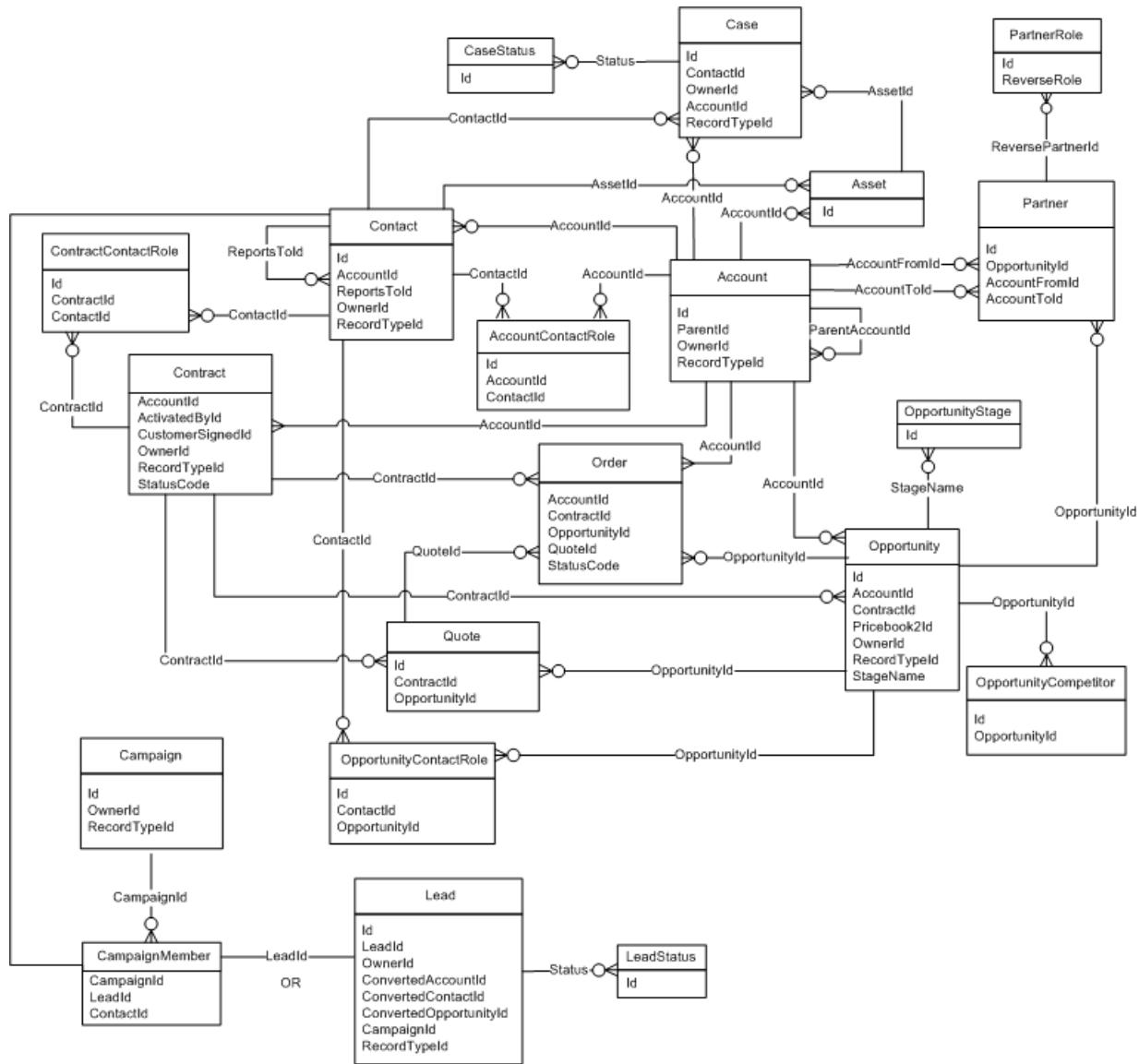
CHAPTER 7 Data Model

The entity relationship diagrams (ERDs) for standard Salesforce objects in this section illustrate important relationships between objects. Salesforce ERDs use crow's foot notation. The following ERDs are available.

- [Sales Objects](#)—includes accounts, contacts, opportunities, leads, campaigns, and other related objects
- [Task and Event Objects](#)—includes tasks and events and their related objects
- [Support Objects](#)—includes cases and solutions and their related objects
- [Salesforce Knowledge Objects](#)—includes view and vote statistics, article versions, and other related objects
- [Document, Note, and Attachment Objects](#)—includes documents, notes, and attachments and their related objects
- [User, Sharing, and Permission Objects](#)—includes users, profiles, and roles
- [Profile and Permission Objects](#)—includes users, profiles, permission sets, and related permission objects
- [Record Type Objects](#)—includes record types and business processes and their related objects
- [Product and Schedule Objects](#)—includes opportunities, products, and schedules
- [Sharing and Team Selling Objects](#)—includes account teams, opportunity teams, and sharing objects
- [Customizable Forecasting Objects](#)—includes forecasts and related objects
- [Forecasts Objects](#)—includes objects for Collaborative Forecasts.
- [Territory Management 2.0 Objects](#)—includes territories and related objects associated with Territory Management 2.0
- [Territory Management](#)—includes territories and related objects
- [Process Objects](#)—includes approval processes and related objects
- [Content Objects](#)—includes content and libraries and their related objects
- [Chatter Feed Objects](#)—includes objects related to feeds
- [Work.com Badge and Reward Objects](#)—includes badge and reward objects
- [Work.com Feedback and Performance Cycle Objects](#)—includes feedback and performance cycle objects

Each ERD includes links to the topics that describe the fields in objects related to the diagram. The data model for your custom objects depends on what you create.

Sales Objects

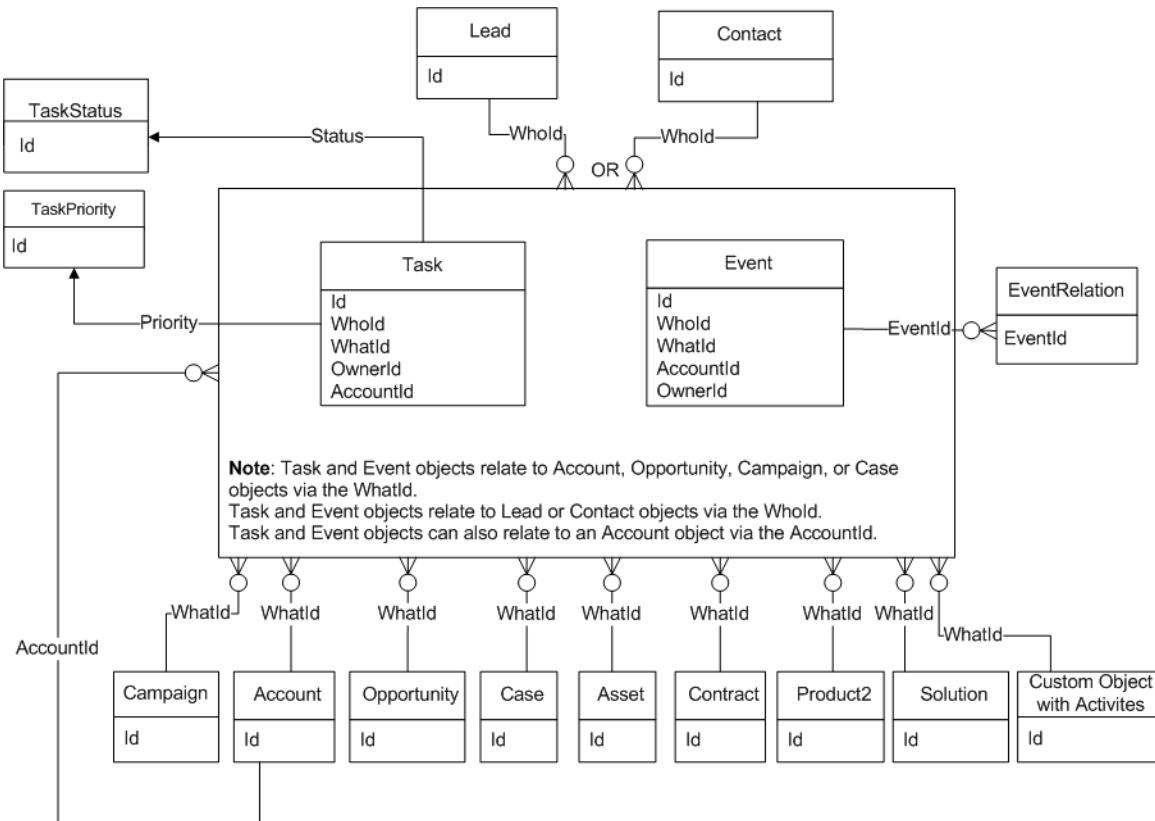


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Task and Event Objects

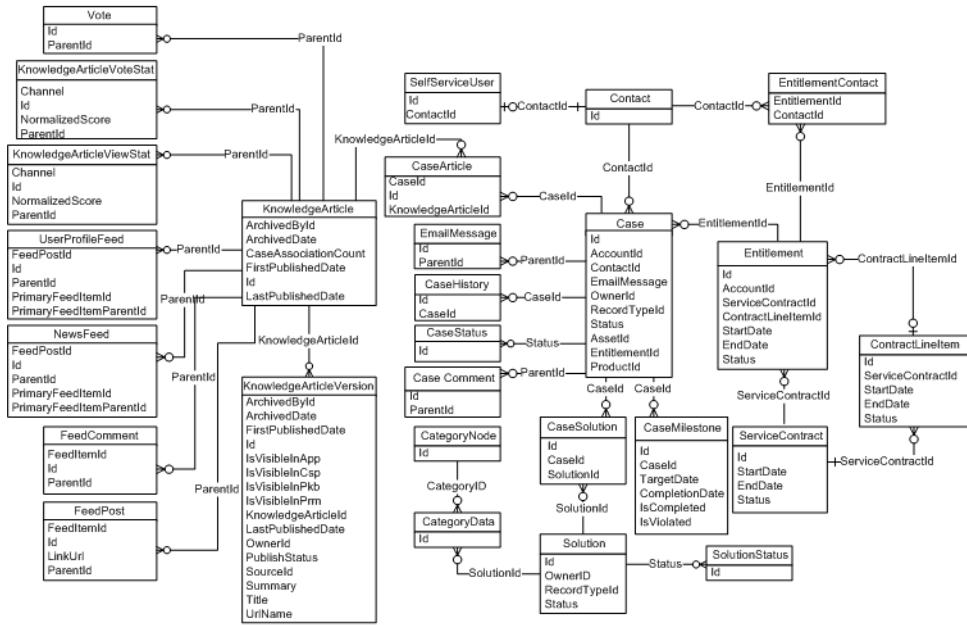


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Support Objects

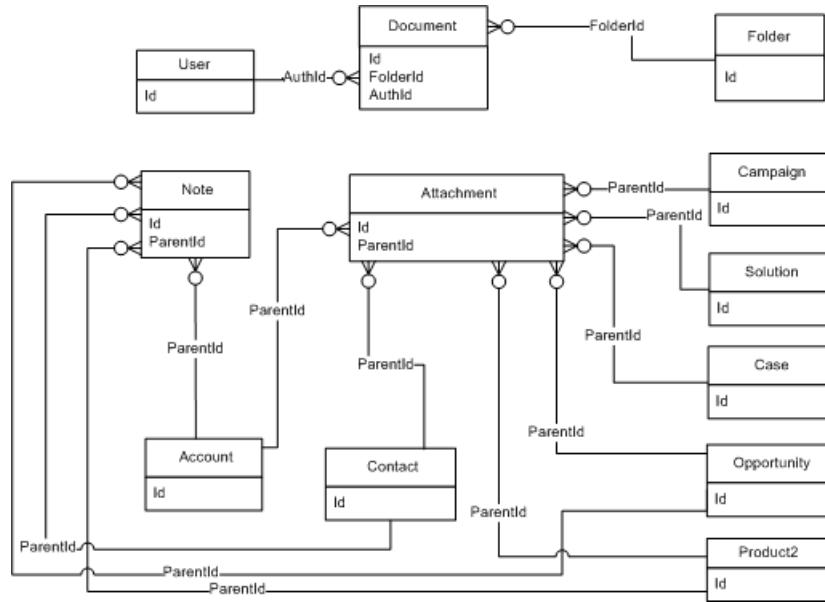


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Document, Note, and Attachment Objects

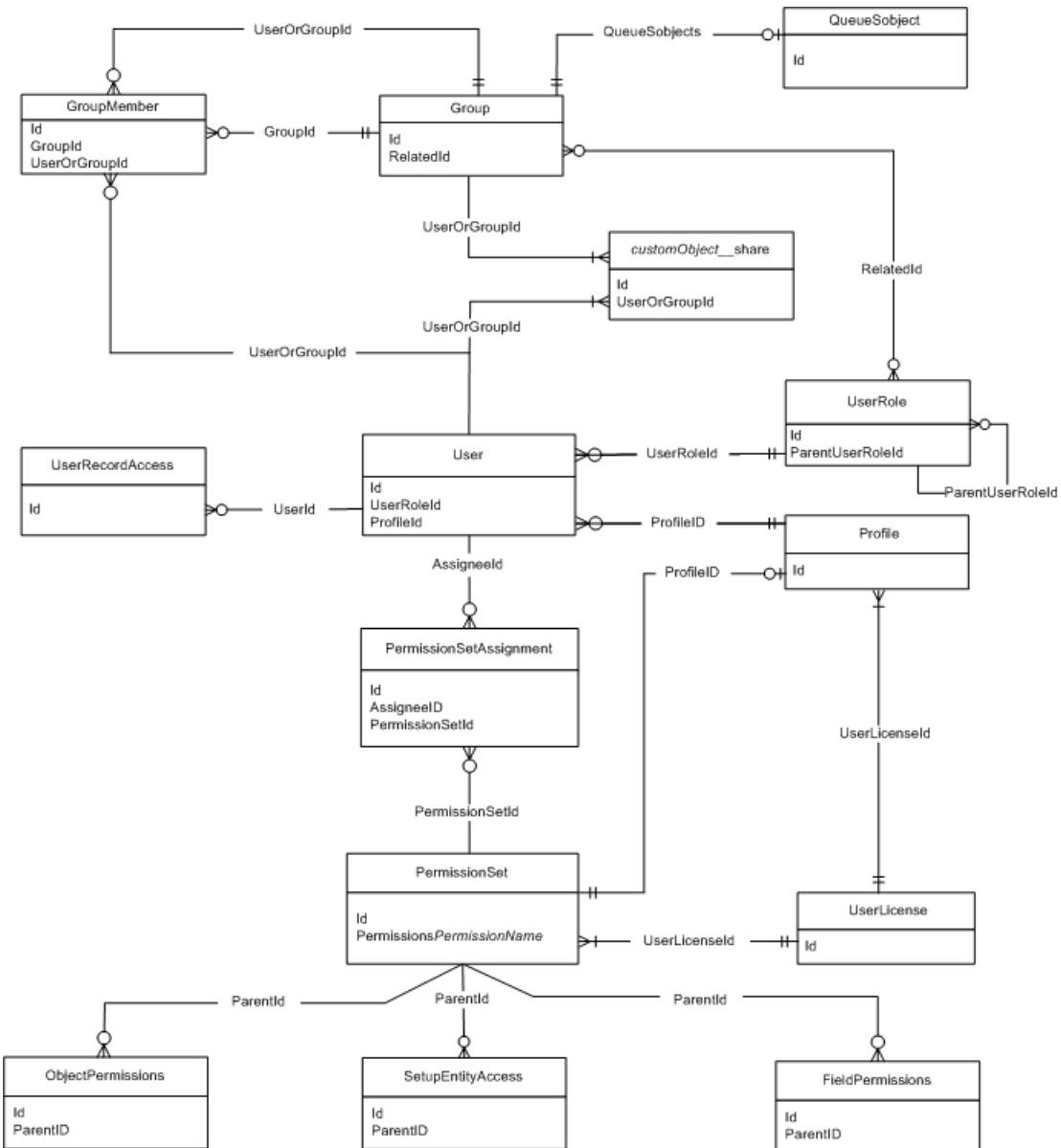


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

User, Sharing, and Permission Objects



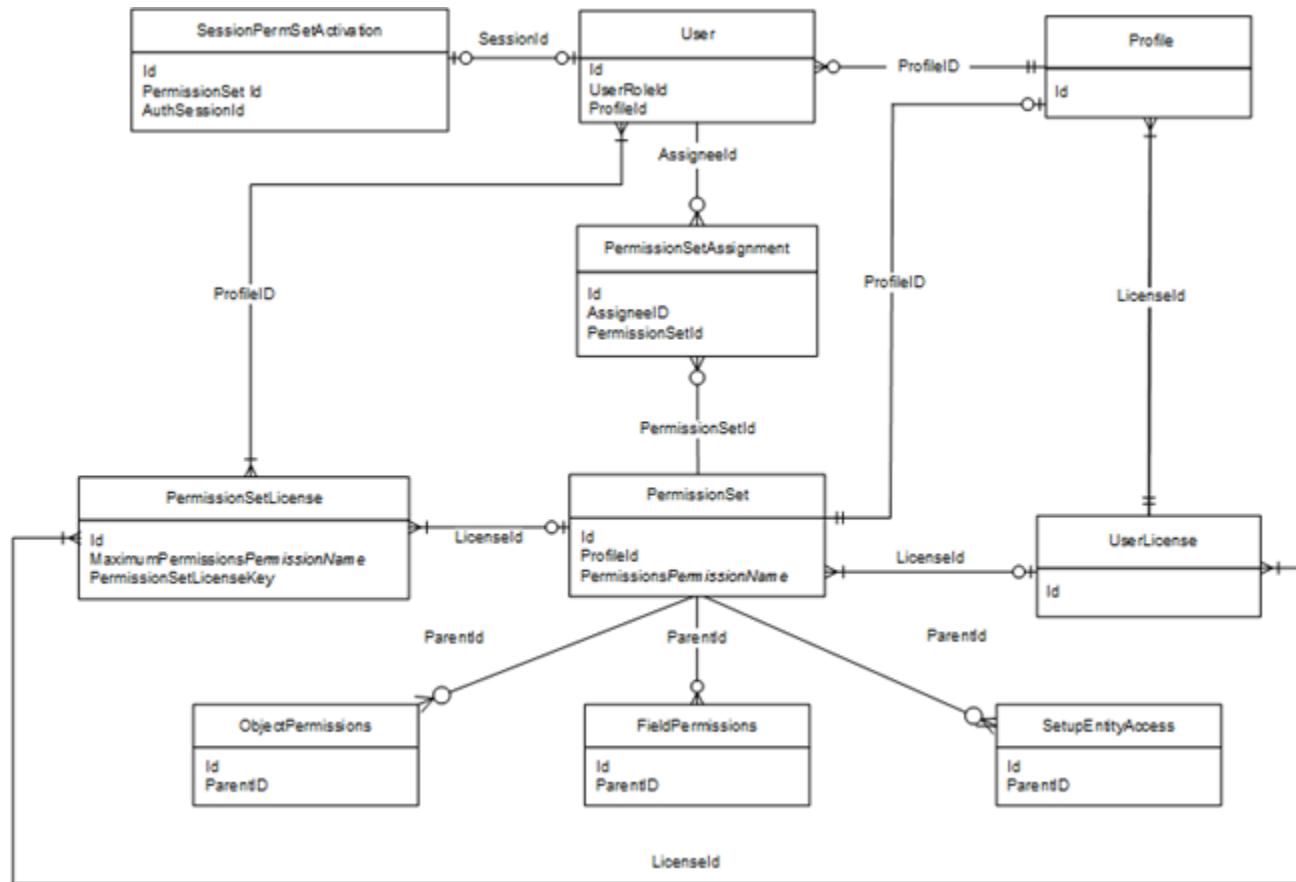
SEE ALSO:

[Standard Objects](#)

[Data Model](#)

[Profile and Permission Objects](#)

Profile and Permission Objects

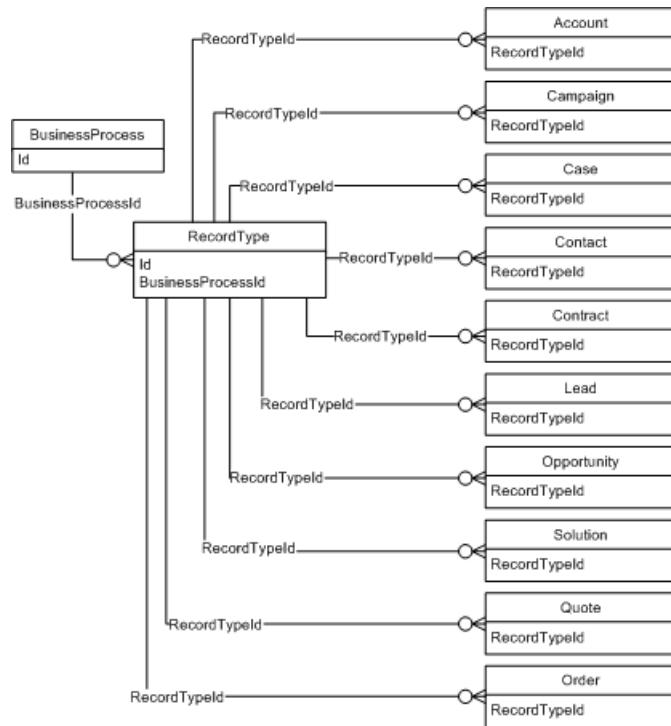


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Record Type Objects

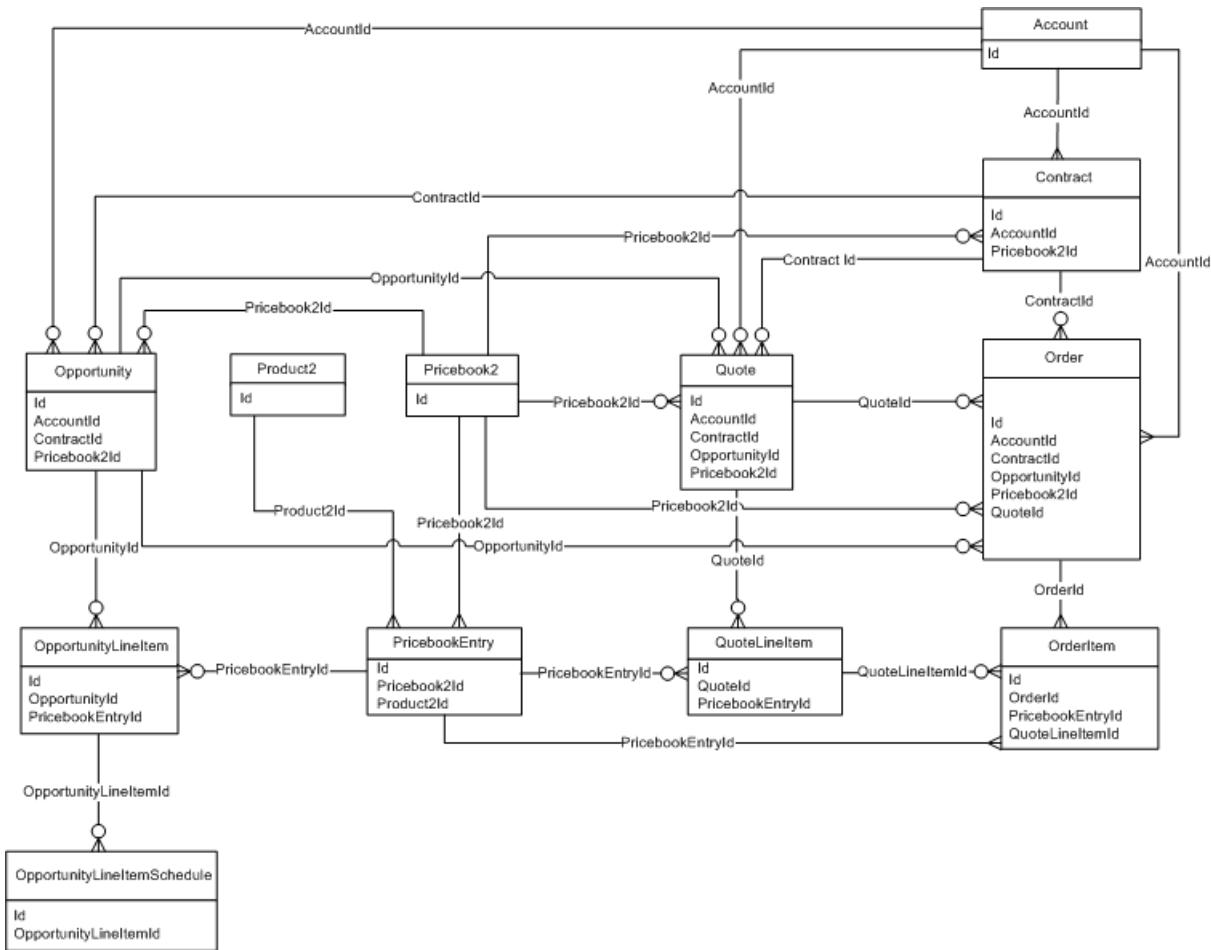


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Product and Schedule Objects



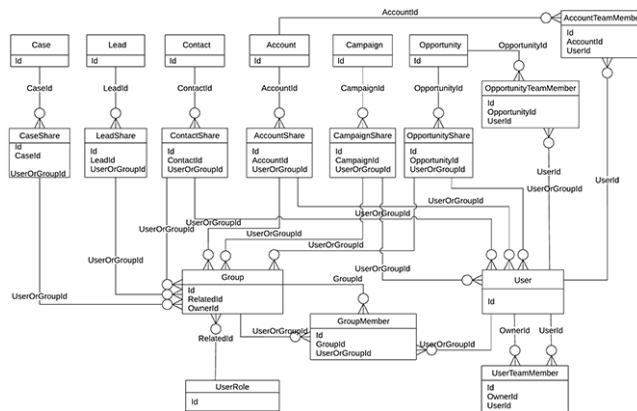
Create a separate PricebookEntry for each currency and price combination.

SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Sharing and Team Selling Objects

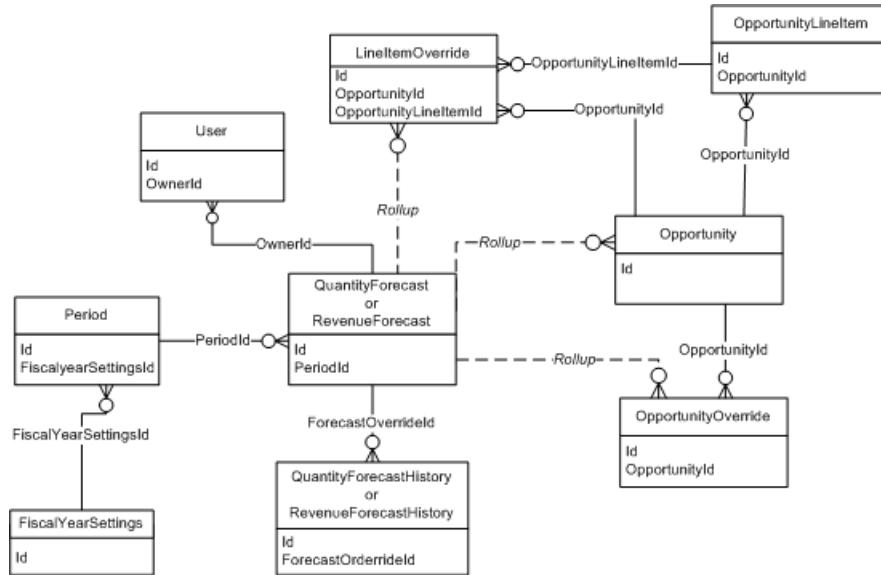


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Customizable Forecasting Objects



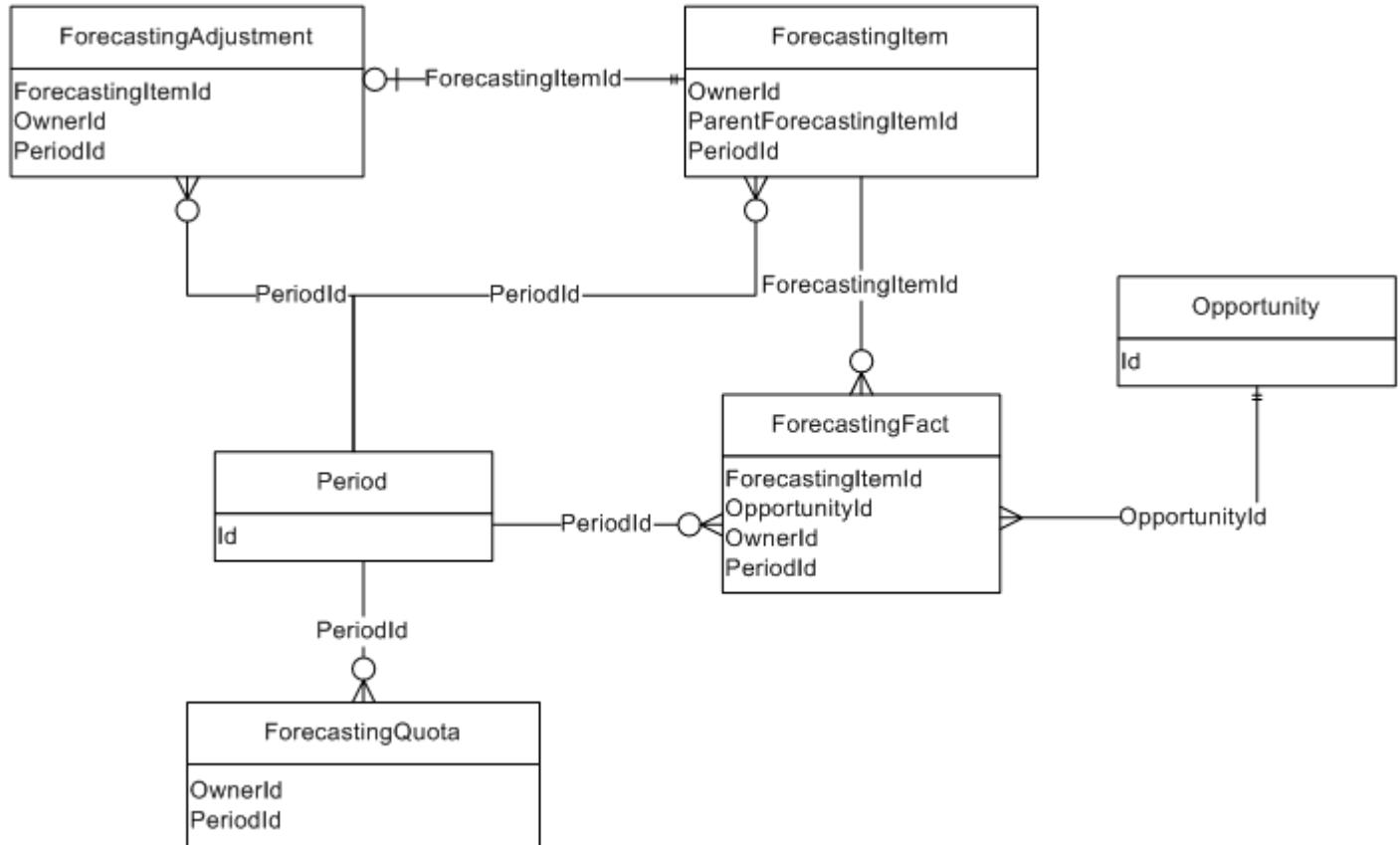
SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Forecasts Objects

 **Note:** This information only applies to Collaborative Forecasts.



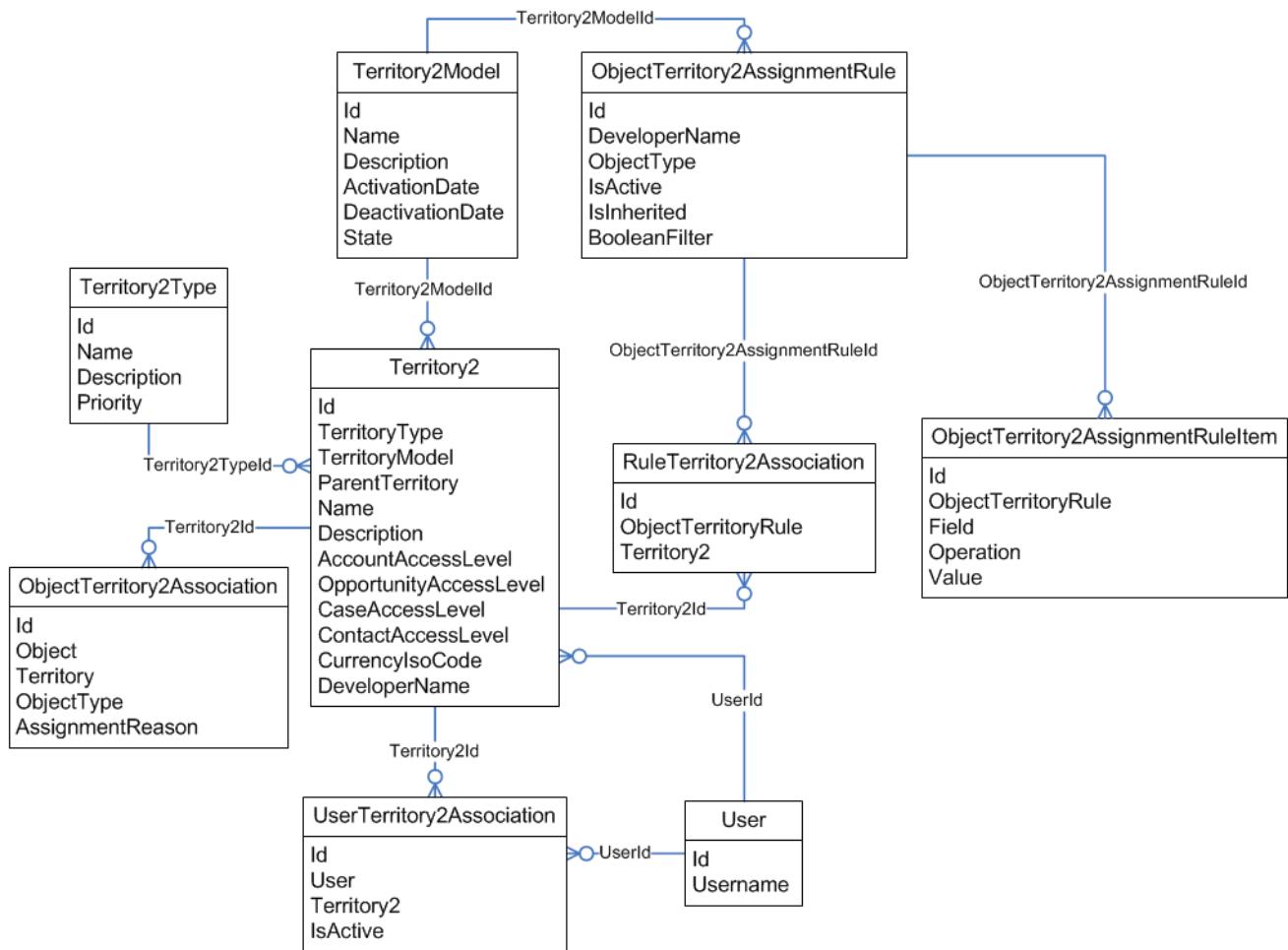
SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Territory Management 2.0 Objects

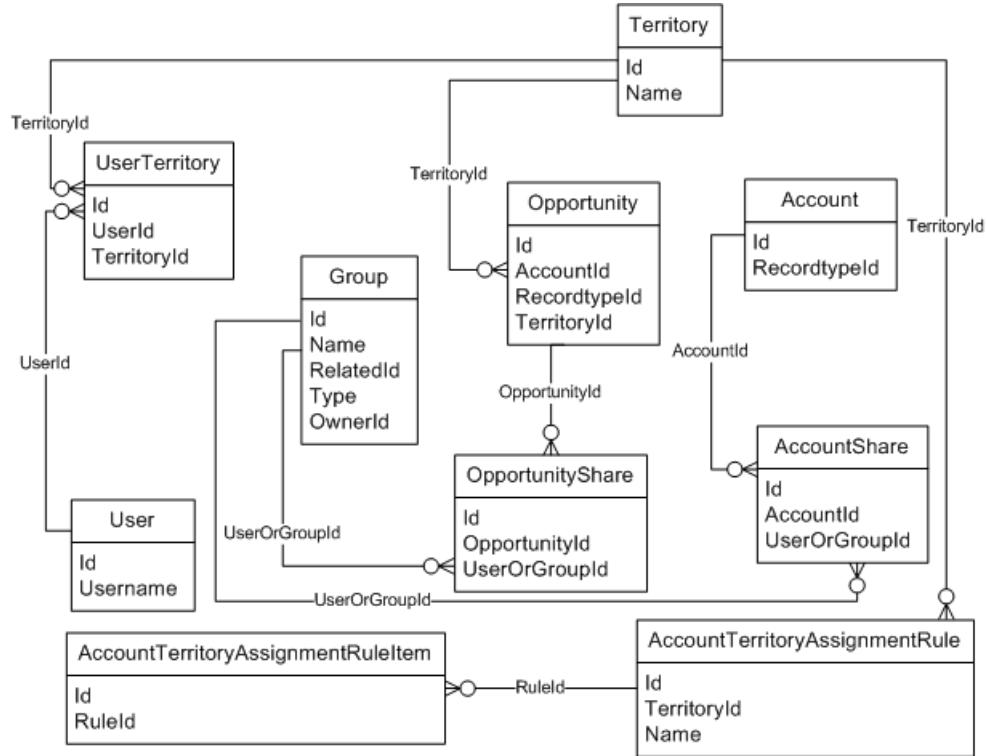
 **Note:** This information applies to Territory Management 2.0 only, not to previous versions of Territory Management.



SEE ALSO:

[Standard Objects](#)[Data Model](#)

Territory Management

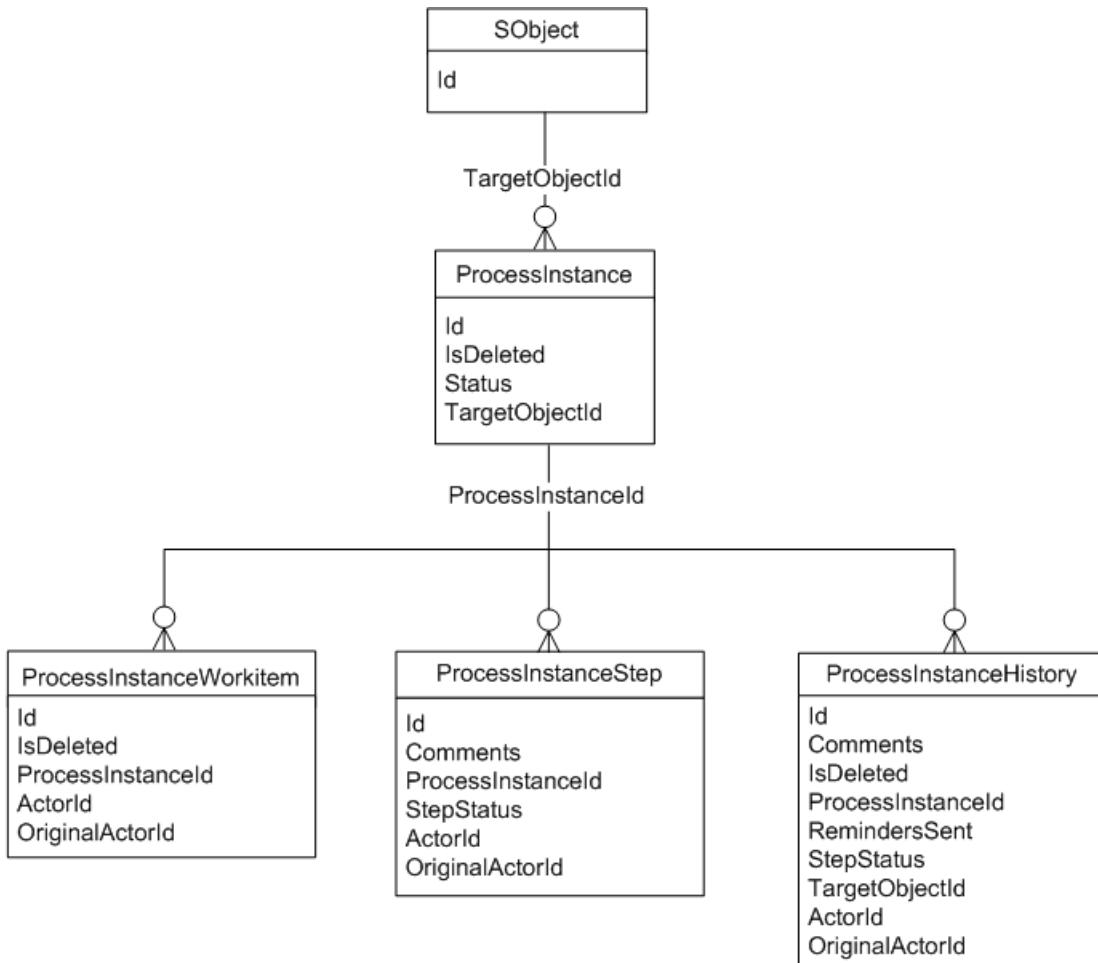


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

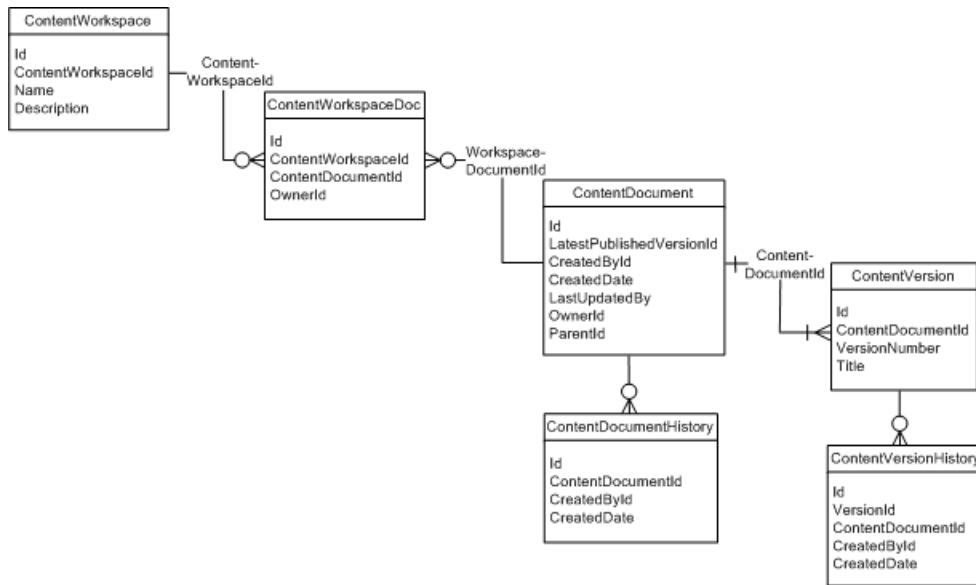
Process Objects



SEE ALSO:

- [Standard Objects](#)
- [Data Model](#)
- [Core Data Types Used in API Calls](#)

Content Objects



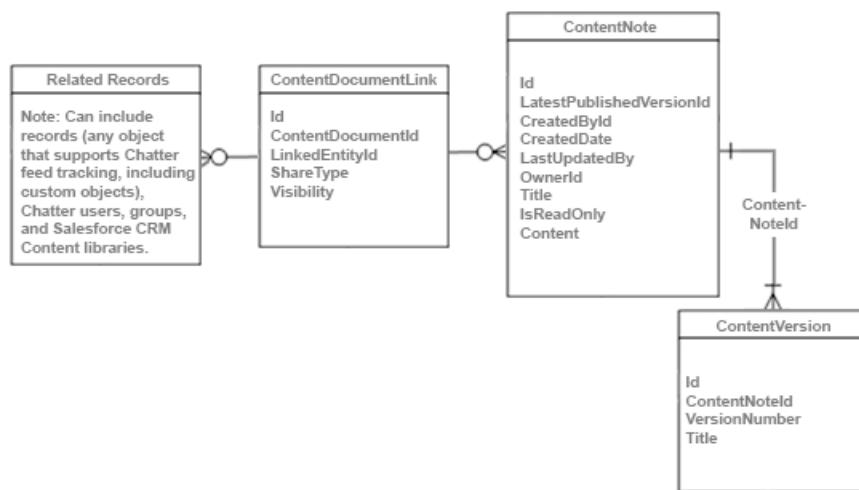
SEE ALSO:

[Standard Objects](#)

[Data Model](#)

ContentNote Objects

The ContentNote object represents notes created with the enhanced version of the Salesforce note-taking tool.



SEE ALSO:

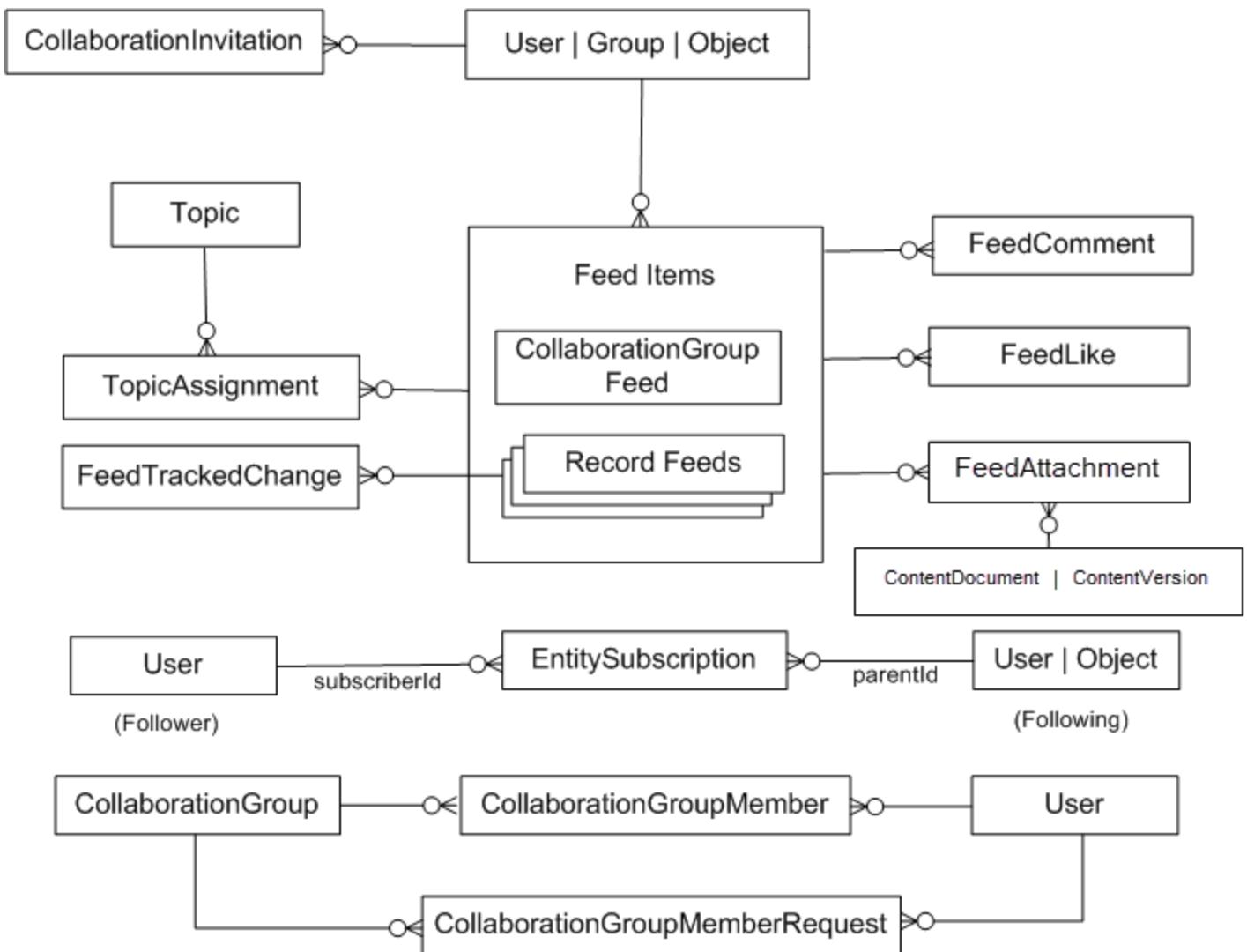
[ContentNote](#)

Chatter Objects

Diagram showing the relationships between the Chatter objects

The following diagram shows the relationships between the major Chatter objects.

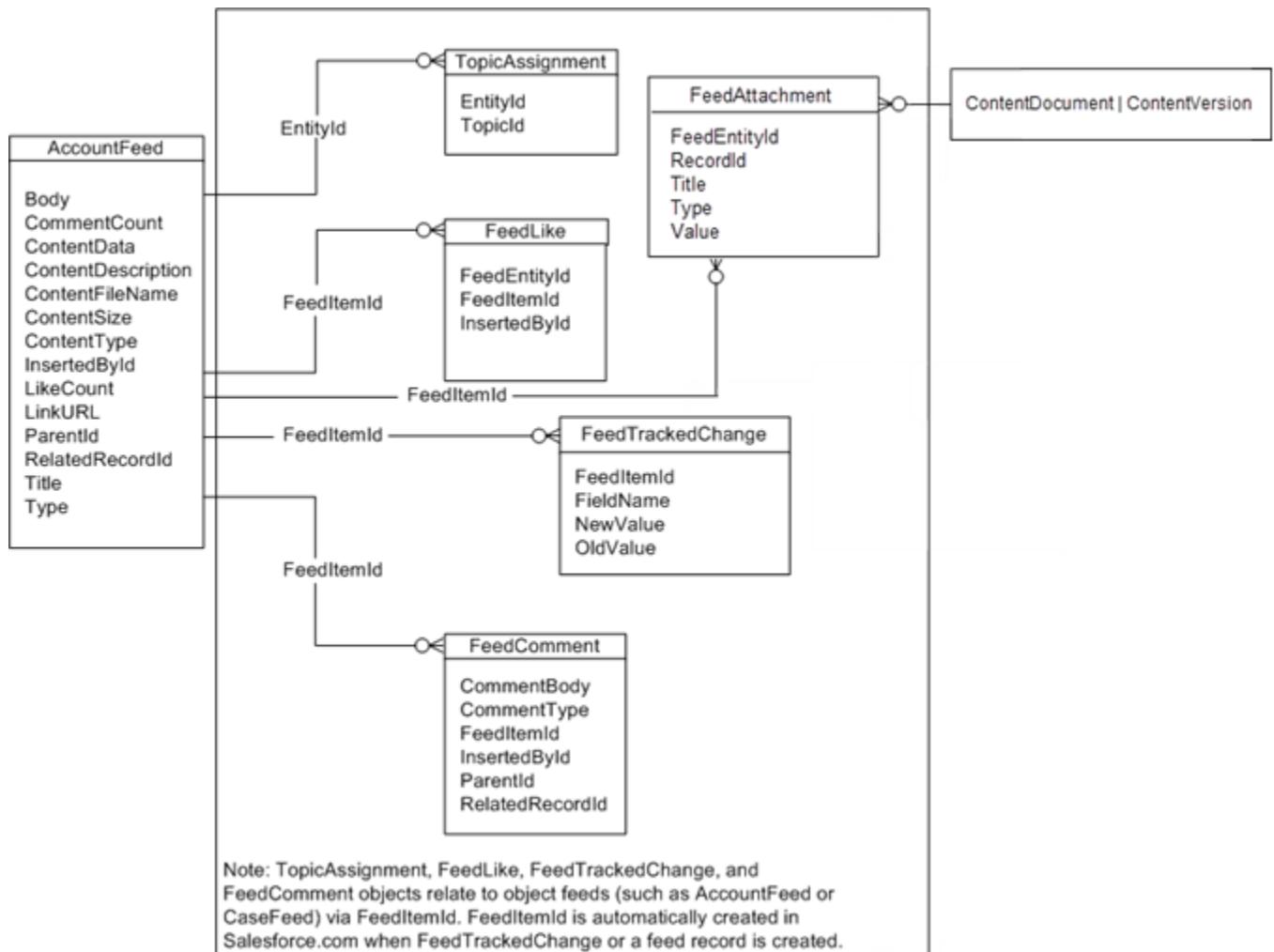
- A feed item is an entry in the feed, such as a change to a record that's being followed, an updated post, or a user status change.
- All feed items have a `ParentId`, which is either:
 - a record
 - a user
 - a group



SEE ALSO:

[Standard Objects](#)[Data Model](#)

Chatter Feed Objects



Standard Objects with Feeds

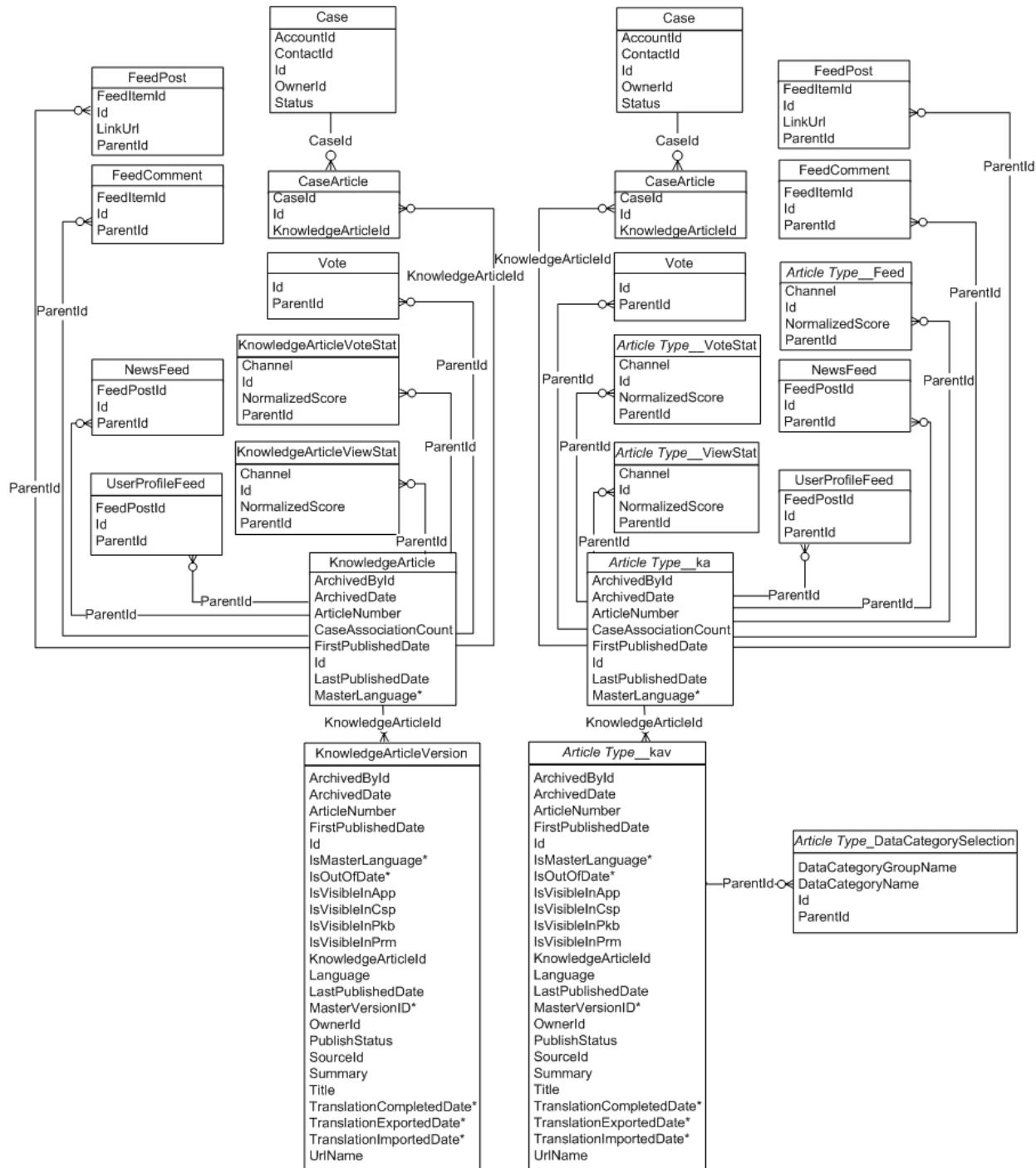
The following standard objects can have Chatter feeds: [Account](#), [Asset](#), [Case](#), [CollaborationGroup](#), [Contact](#), [ContentDocument](#), [Contract](#), [Dashboard](#), [DashboardComponent](#), [Entitlement](#), [Event](#), [KnowledgeArticle](#), [Lead](#), [Opportunity](#), [Product2](#), [Quote](#), [Report](#), [ServiceContract](#), [Site](#), [Solution](#), [Task](#), [Topic](#), [User](#), [WorkOrder](#), and [WorkOrderLineItem](#).

SEE ALSO:

[Standard Objects](#)

[Data Model](#)

Salesforce Knowledge Objects

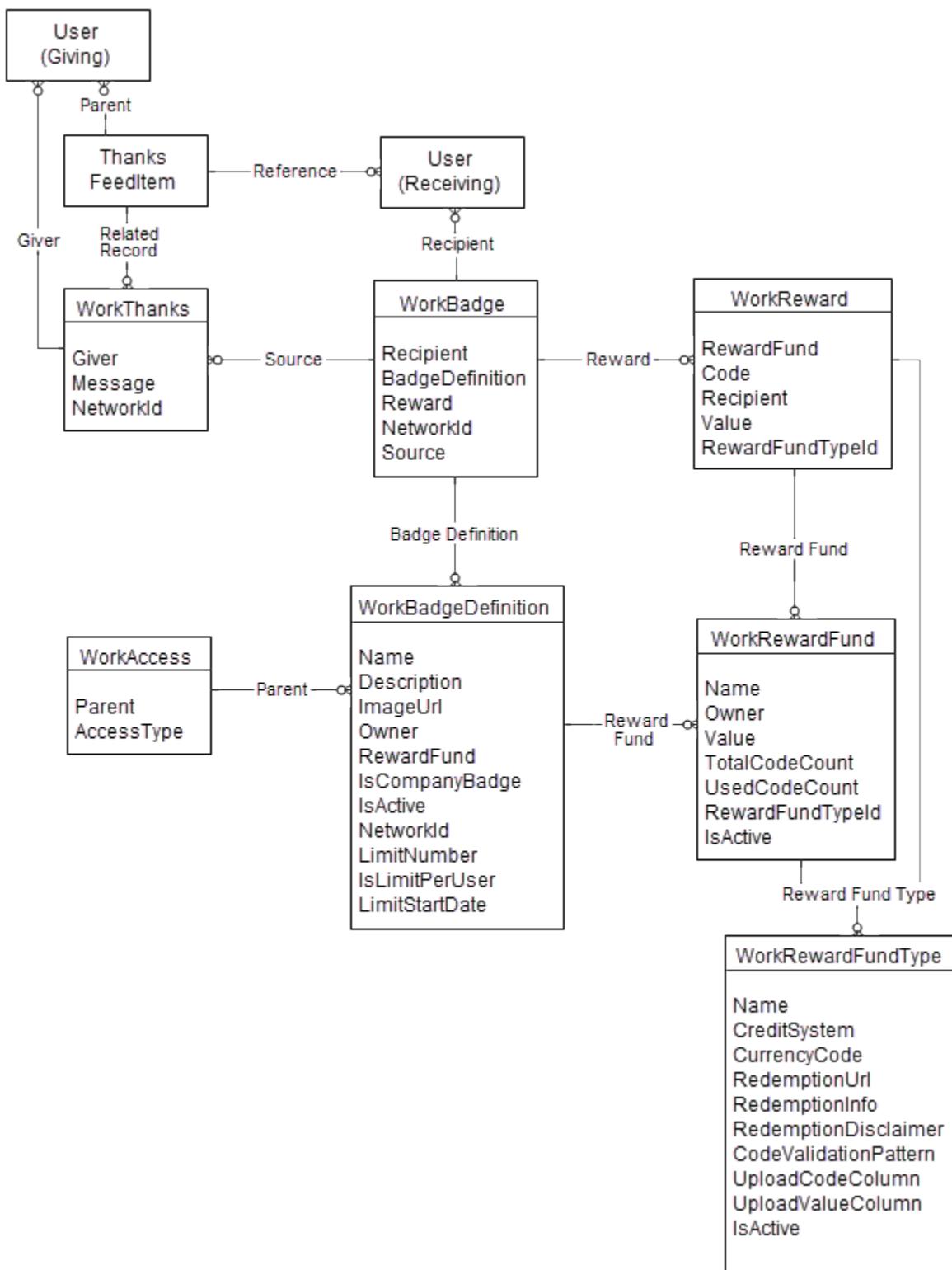


SEE ALSO:

[Standard Objects](#)

[Data Model](#)

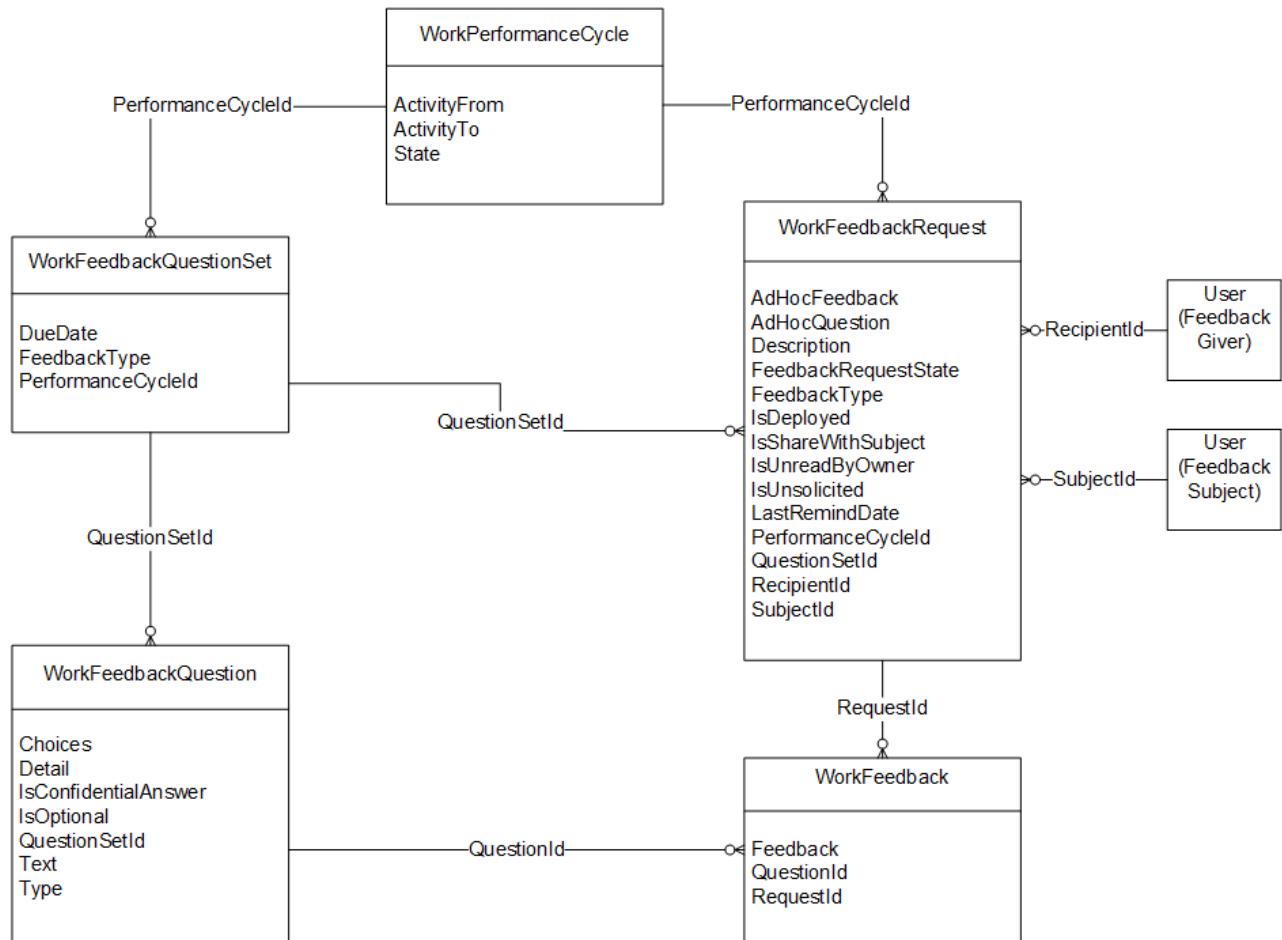
Work.com Badge and Reward Objects



SEE ALSO:

[Standard Objects](#)
[Data Model](#)

Work.com Feedback and Performance Cycle Objects



SEE ALSO:

[Standard Objects](#)
[Data Model](#)

CHAPTER 8 Standard Objects

This section provides a list of standard objects and their standard fields. Some fields may not be listed for some objects.

To see the system fields for each object, see [System Fields](#).

To verify the complete list of fields for an object, you can use a describe call from the API, or inspect with an appropriate tool, for example, inspecting the WSDL or using a schema viewer.

[AcceptedEventRelation](#)

Represents invitees with the status Accepted for a given event.

[Account](#)

Represents an individual account, which is an organization or person involved with your business (such as customers, competitors, and partners).

[AccountContactRelation](#)

Represents a relationship between a contact and one or more accounts.

[AccountCleanInfo](#)

Stores the metadata Data.com Clean uses to determine an account record's clean status. AccountCleanInfo helps you automate the cleaning or related processing of account records.

[AccountContactRole](#)

Represents the role that a Contact plays on an Account.

[AccountFeed](#)

Represents a single feed item on an account record detail page. This object is available in API version 18.0 and later.

[AccountHistory](#)

Represents the history of changes to the values in the fields of an account. This object is available in versions 11.0 and later.

[AccountOwnerSharingRule](#)

Represents the rules for sharing an account with a User other than the owner.

[AccountPartner](#)

This read-only object represents a partner relationship between two Account objects. It is automatically created when a Partner object is created for a partner relationship between two accounts.

[AccountShare](#)

Represents a sharing entry on an Account.

[AccountTag](#)

Associates a word or short phrase with an Account.

[AccountTeamMember](#)

Represents a User who is a member of an Account team.

Standard Objects

[AccountTerritoryAssignmentRule](#)

An account assignment rule that assigns accounts to territories based on account fields. Only available if territory management has been enabled for your organization.

[AccountTerritoryAssignmentRuleItem](#)

A row of selection criteria for an AccountTerritoryAssignmentRule object. Only available if territory management has been enabled for your organization.

[AccountTerritorySharingRule](#)

Represents the rules for sharing an Account within a Territory.

[ActionLinkGroupTemplate](#)

Action link templates let you reuse action link definitions and package and distribute action links. An action link is a button on a feed element. Clicking on an action link can take a user to another Web page, initiate a file download, or invoke an API call to an external server or Salesforce. Use action links to integrate Salesforce and third-party services into the feed. Every action link belongs to an action link group and action links within the group are mutually exclusive. This object is available in API version 33.0 and later.

[ActionLinkTemplate](#)

Action link templates let you reuse action link definitions and package and distribute action links. An action link is a button on a feed element. Clicking on an action link can take a user to another Web page, initiate a file download, or invoke an API call to an external server or Salesforce. Use action links to integrate Salesforce and third-party services into the feed. This object is available in API version 33.0 and later.

[ActivityHistory](#)

This read-only object is displayed in a related list of closed activities—past events and closed tasks—related to an object. It includes activities for all contacts related to the object. ActivityHistory fields for phone calls are only available if your organization uses Salesforce CRM Call Center.

[AdditionalNumber](#)

Represents an optional additional number for a call center. This additional number is visible in the call center's phone directory.

[AgentWork](#)

Represents a work assignment that's been routed to an agent. This object is available in API version 32.0 and later.

[AllowedEmailDomain](#)

Represents an allowed email domain for users in your organization. You can define a whitelist to restrict the email domains allowed in a user's Email field. This object is available in API version 29.0 and later.

[Announcement](#)

Represents a Chatter group announcement. This object is available in API version 30.0 and later.

[ApexClass](#)

Represents an Apex class.

[ApexComponent](#)

Represents a definition for a custom component that can be used in a Visualforce page alongside standard components such as `<apex:relatedList>` and `<apex:dataTable>`.

[ApexLog](#)

Represents a debug log, containing information about a transaction, including information about Apex, Visualforce, workflow and validation rules. This object is available in API version 19.0 and later.

[ApexPage](#)

Represents a single Visualforce page.

[ApexTestQueueItem](#)

Represents a single Apex class in the Apex job queue. This object is available in API version 23.0 and later.

Standard Objects

[ApexTestResult](#)

Represents the result of an Apex test method execution. This object is available in API version 23.0 and later.

[ApexTestResultLimits](#)

Captures the Apex test limits used for a particular test method execution. An instance of this object is associated with each ApexTestResult record. This object is available in API version 37.0 and later.

[ApexTestRunResult](#)

Contains summary information about all the test methods that were run in a particular Apex job. This object is available in API version 37.0 and later.

[ApexTestSuite](#)

Represents a suite of Apex classes to include in a test run. A TestSuiteMembership object associates each class with the suite. This object is available in API version 36.0 and later.

[ApexTrigger](#)

Represents an Apex trigger.

[AppMenuItem](#)

Represents the organization's default settings for items in the Force.com app menu or App Launcher.

[Approval](#)

Represents an approval request for a Contract.

[Article Type__DataCategorySelection](#)

A data category selection represents a data category that classifies an article. This object is available in API version 19.0 and later.

[Article Type__Feed](#)

Represents a single feed item in the feed displayed on the detail page for an article. This object is available in API version 20.0 and later.

[Asset](#)

Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased and installed.

[AssetFeed](#)

Represents a single feed item in the feed displayed on the detail page for an asset record. This object is available in API version 18.0 and later.

[AssetOwnerSharingRule](#)

Represents the rules for sharing an Asset with users other than the owner. This object is available in API version 33.0 and later.

[AssetShare](#)

Represents a sharing entry on an Asset. This object is available in API version 33.0 and later.

[AssetTag](#)

Associates a word or short phrase with an Asset.

[AssignedResource](#)

Represents a service resource who is assigned to a service appointment. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

[AssignmentRule](#)

Represents an assignment rule associated with a Case or Lead.

[AsyncApexJob](#)

Represents an individual Apex sharing recalculation job, a batch Apex job, a method with the `future` annotation, or a job that implements `Queueable`.

Standard Objects

[AttachedContentDocument](#)

This read-only object contains all `ContentDocument` objects associated with an object.

[AttachedContentNote](#)

This read-only object contains all `ContentNote` objects associated with an object. This object is available in API version 35.0 and later.

[Attachment](#)

Represents a file that a User has uploaded and attached to a parent object.

[AuraDefinition](#)

Represents a Lightning definition, such as component markup, a client-side controller, or an event. This object is available in API version 32.0 and later.

[AuraDefinitionBundle](#)

Represents a Lightning definition bundle, such as a component or application bundle. A bundle contains a Lightning definition and all its related resources. This object is available in API version 32.0 and later.

[AuthConfig](#)

Represents authentication options for a community or custom domain that was created by using My Domain. This object is available in API version 32.0 and later.

[AuthConfigProviders](#)

Represents an authentication provider that's configured in an organization. This object is a child of the `AuthConfig` object. This object is available in API version 32.0 and later.

[AuthProvider](#)

Represents an authentication provider (or auth provider) in your organization.

[AuthSession](#)

The `AuthSession` object represents an individual user session in your organization. This object is available in versions 29.0 and later.

[BackgroundOperation](#)

Represents a background operation in an asynchronous job queue. This object is available in API version 35.0 and later.

[BackgroundOperationResult](#)

Stores error messages generated when running Async SOQL queries or importing data into BigObjects using Bulk API. This is a `BigObject`, available in API version 37.0 and later.

[Bookmark](#)

Represents a link between opportunities that share common information.

[BrandTemplate](#)

Letterhead for HTML `EmailTemplate`.

[BusinessHours](#)

Specifies the business hours of your support organization. Escalation rules are run only during these hours.

[BusinessProcess](#)

Represents a business process.

[CallCenter](#)

Represents a call center, which is a logical representation of a single computer-telephony integration (CTI) system instance in an organization.

[Campaign](#)

Represents and tracks a marketing campaign, such as a direct mail promotion, webinar, or trade show.

Standard Objects

[CampaignFeed](#)

Represents a single feed item in the feed on a campaign record detail page. This object is available in API version 18.0 and later.

[CampaignInfluence](#)

Represents the association between a campaign and an opportunity in Customizable Campaign Influence. This object is available in API version 37.0 and later.

[CampaignInfluenceModel](#)

This read-only object represents a campaign influence model in Customizable Campaign Influence. Use campaign influence models to group `CampaignInfluence` records created by a specific set of triggers and workflows that you define. The Salesforce influence model is the default model. This object is available in API version 37.0 and later.

[CampaignMember](#)

Represents the association between a campaign and either a lead or a contact.

[CampaignMemberStatus](#)

One or more member status values defined for a campaign.

[CampaignOwnerSharingRule](#)

Represents the rules for sharing a campaign with User records other than the owner or anyone above the owner in the role hierarchy.

[CampaignShare](#)

Represents a sharing entry on a Campaign.

[CampaignTag](#)

Associates a word or short phrase with a Campaign.

[Case](#)

Represents a case, which is a customer issue or problem.

[CaseArticle](#)

Represents the association between a Case and a KnowledgeArticle. This object is available in API version 20.0 and later.

[CaseComment](#)

Represents a comment that provides additional information about the associated Case.

[CaseContactRole](#)

Represents the role that a given Contact plays on a Case.

[CaseFeed](#)

Represents a single feed item in the feed displayed on the detail page for a case record. A case feed shows recent changes to a case record for any fields that are tracked in feeds, and comments and posts about the record. It is a useful way to stay up-to-date with changes to cases. This object is available in API version 18.0 and later.

[CaseHistory](#)

Represents historical information about changes that have been made to the associated Case.

[CaseMilestone](#)

Represents a milestone (required step in a customer support process) on a Case. This object is available in API version 18.0 and later.

[CaseOwnerSharingRule](#)

Represents the rules for sharing a case with users other than the owner.

[CaseShare](#)

Represents a sharing entry on a Case.

[CaseSolution](#)

Represents the association between a Case and a Solution.

Standard Objects

[CaseStatus](#)

Represents the status of a Case, such as New, On Hold, or In Process.

[CaseTag](#)

Associates a word or short phrase with a Case

[CaseTeamMember](#)

Represents a case team member, who works with a team of other users to help resolve a case.

[CaseTeamRole](#)

Represents a case team role. Every case team member has a role on a case, such as "Customer Contact" or "Case Manager."

[CaseTeamTemplate](#)

Represents a predefined case team, which is a group of users that helps resolve a case.

[CaseTeamTemplateMember](#)

Represents a member on a predefined case team, which is a group of users that helps resolve cases.

[CaseTeamTemplateRecord](#)

The CaseTeamTemplateRecord object is a linking object between the Case and CaseTeamTemplate objects. To assign a predefined case team to a case (customer inquiry), create a CaseTeamTemplateRecord record and point the ParentId to the case and the TeamTemplateId to the predefined case team.

[CategoryData](#)

Represents a logical grouping of Solution records.

[CategoryNode](#)

Represents a tree of Solution categories.

[CategoryNodeLocalization](#)

When the Translation Workbench is enabled for your organization, the CategoryNodeLocalization object provides the translation of the label of a solution category.

[ChatterActivity](#)

ChatterActivity represents the number of posts and comments made by a user and the number of comments and likes on posts and comments received by the same user. This object is available in API version 23.0 and later.

[ChatterAnswersActivity](#)

Represents the reputation of a User in Chatter Answers communities. This object is available in API version 25.0 and later.

[ChatterAnswersReputationLevel](#)

Represents a reputation level within a Chatter Answers zone. This object is available in API version 26.0 and later.

[ChatterConversation](#)

Represents a private conversation in Chatter, consisting of messages that conversation members have sent or received. This object is available in API version 23.0 and later.

[ChatterConversationMember](#)

Represents a member of a private conversation in Chatter. A member has either sent messages to or received messages from other conversation participants. This object is available in API version 23.0 and later.

[ChatterMessage](#)

Represents a message sent as part of a private conversation in Chatter. This object is available in API version 23.0 and later.

[ClientBrowser](#)

Represents a cookie added to the browser upon login, and also includes information about the browser application where the cookie was inserted. This object is available in version 28.0 and later.

Standard Objects

[CollaborationGroup](#)

Represents a Chatter group. This object is available in API version 19.0 and later.

[CollaborationGroupFeed](#)

Represents a single feed item on a Chatter group feed. A group feed shows posts and comments about the group. This object is available in API version 19.0 and later.

[CollaborationGroupMember](#)

Represents a member of a Chatter group. This object is available in API version 19.0 and later.

[CollaborationGroupMemberRequest](#)

Represents a request to join a private Chatter group. This object is available in API version 21.0 and later.

[CollaborationGroupRecord](#)

Represents the records associated with Chatter groups.

[CollaborationInvitation](#)

Represents an invitation to join Chatter, either directly or through a group. This object is available in API version 21.0 and later.

[CombinedAttachment](#)

This read-only object contains all notes, attachments, Google Docs, documents uploaded to libraries in Salesforce CRM Content, and files added to Chatter that are associated with a record.

[Community \(Zone\)](#)

Represents a zone that contains Idea or Question objects.

[ConnectedApplication](#)

Represents a connected app and its details; all fields are read-only.

[Contact](#)

Represents a contact, which is an individual associated with an account.

[ContactCleanInfo](#)

Stores the metadata Data.com Clean uses to determine a contact record's clean status. Helps you automate the cleaning or related processing of contact records. ContactCleanInfo includes a number of bit vector fields.

[ContactFeed](#)

Represents a single feed item in the feed on a contact record detail page. This object is available in API version 18.0 and later.

[ContactHistory](#)

Represents the history of changes to the values in the fields of a contact. This object is available in versions 11.0 and later.

[ContactOwnerSharingRule](#)

Represents the rules for sharing a contact with a User other than the owner.

[ContactShare](#)

Represents a list of access levels to a Contact along with an explanation of the access level. For example, if you have access to a record because you own it, the ContactAccessLevel is All and RowCause is Owner.

[ContactTag](#)

Associates a word or short phrase with a Contact.

[ContentAsset](#)

Represents a Salesforce file that has been converted to an asset file in a custom app inLightning Experience. Enables a Salesforce file to be used for org setup and configuration purposes. Asset files can be packaged and referenced by other components. This object is available in API version 38.0 and later.

Standard Objects

[ContentDistribution](#)

Represents information about sharing a document externally. This object is available in API version 32.0 and later.

[ContentDistributionView](#)

Represents information about views of a shared document. This read-only object is available in API version 32.0 and later.

[ContentDocument](#)

Represents a document that has been uploaded to a library in Salesforce CRM Content or Salesforce Files. This object is available in API versions 17.0 and later for Salesforce CRM Content. This object is available in API version 21.0 and later for Salesforce Files.

[ContentDocumentFeed](#)

Represents a single feed item associated with ContentDocument. A content document feed shows these content document changes: creating a ContentDocument file and uploading a new ContentDocument. This object is available in versions 20.0 and later.

[ContentDocumentHistory](#)

Represents the history of a document. This object is available in versions 17.0 and later.

[ContentDocumentLink](#)

Represents the link between a Salesforce CRM Content document or Salesforce file and where it's shared. A file can be shared with other users, groups, records, and Salesforce CRM Content libraries. This object is available in versions 21.0 and later for Salesforce CRM Content documents and Salesforce Files.

[ContentFolder](#)

Represents a folder in a content library for adding files. This object is available in API version 34.0 and later.

[ContentFolderItem](#)

Represents a file (ContentDocument) or folder (ContentFolder) that resides in a ContentFolder in a ContentWorkspace. This object is available in API version 35.0 and later.

[ContentFolderLink](#)

Defines the association between a library and its root folder. This object is available in API version 34.0 and later.

[ContentFolderMember](#)

Defines the association between a file and a folder. This object is available in API version 34.0 and later.

[ContentHubItem](#)

Represents a file or folder in a Files Connect external data source, such as Microsoft SharePoint or OneDrive for Business. This object is available in API version 33.0 and later.

[ContentHubRepository](#)

Represents a Files Connect external data source such as Microsoft SharePoint or OneDrive for Business. This object is available in API version 33.0 and later.

[ContentNote](#)

Represents a note in Salesforce. This object is available in API version 32.0 and later.

[ContentVersion](#)

Represents a specific version of a document in Salesforce CRM Content or Salesforce Files. This object is available in versions 17.0 and later for Salesforce CRM Content documents. This object is available in versions 20.0 and later for Salesforce Files.

[ContentVersionHistory](#)

Represents the history of a specific version of a document. This object is available in version 17.0 and later.

[ContentWorkspace](#)

Represents a public library in Salesforce CRM Content. This object is available in versions 17.0 and later.

Standard Objects

[ContentWorkspaceDoc](#)

Represents a link between a document and a public library in Salesforce CRM Content. This object is available in versions 17.0 and later.

[Contract](#)

Represents a contract (a business agreement) associated with an Account.

[ContractContactRole](#)

Represents the role that a given Contact plays on a Contract.

[ContractFeed](#)

Represents a single feed item in the feed on the contract record detail page. This object is available in API version 18.0 and later.

[ContractHistory](#)

Represents the history of changes to the values in the fields of a contract.

[ContractLineItem](#)

Represents a Product2 in a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

[ContractLineItemHistory](#)

Represents the history of changes to the values in the fields on a ContractLineItem (items in a customer support agreement). This object is available in API version 18.0 and later.

[ContractStatus](#)

Represents the status of a Contract, such as Draft, InApproval, Activated, Terminated, or Expired.

[ContractTag](#)

Associates a word or short phrase with a Contract.

[CronJobDetail](#)

Contains details about the associated scheduled job, such as the job's name and type. This object is available in API version 29.0 and later.

[CronTrigger](#)

Contains schedule information for a scheduled job. CronTrigger is similar to a cron job on UNIX systems. This object is available in API version 17.0 and later.

[CurrencyType](#)

Represents the currencies used by an organization for which the multicurrency feature is enabled.

[CustomBrand](#)

Represents a custom branding and color scheme. This object is available in API version 28.0 and later.

[CustomBrandAsset](#)

Represents a branding element in a custom branding scheme. For example, a color, logo image, header image, or footer text. A CustomBrandAsset can apply to a community or to an organization using Salesforce1. This object is available in API version 28.0 and later.

[Custom Metadata Type__mdt](#)

Represents a custom metadata record. This object is available in API version 34.0 and later.

[Custom Object__Feed](#)

Represents a single feed item on a custom object detail page. This object is available in API version 18.0 and later.

[CustomPermission](#)

Represents a permission created to control access to a custom process or app, such as sending email. This object is available in API version 31.0 and later.

Standard Objects

[CustomPermissionDependency](#)

Represents the dependency between two custom permissions when one custom permission requires that you enable another custom permission. This object is available in API version 32.0 and later.

[DandBCompany](#)

Represents a Dun & Bradstreet® company record, which is associated with an account added from Data.com. This object is available in API version 25.0 and later.

[Dashboard](#)

Represents a dashboard, which shows data from custom reports as visual components. Access is read-only. This object is available in API version 20.0 and later.

[DashboardComponent](#)

Represents a dashboard component, which can be a chart, metric, table, or gauge on a dashboard. Access is read-only. This object is available in API version 21.0 and later.

[DashboardComponentFeed](#)

Represents a single feed item in the feed displayed on a dashboard component. This object is available in API version 21.0 and later.

[DashboardFeed](#)

Represents a single feed item in the feed displayed on a dashboard. This object is available in API version 20.0 and later.

[DashboardTag](#)

Associates a word or short phrase with a Dashboard. This object is available in API version 20.0 and later.

[DataAssessmentFieldMetric](#)

Represents summary statistics for matched, blank, and differing fields in account records of an org compared to records in Data.com. This object is available in API version 37.0 and later.

[DataAssessmentMetric](#)

Represents a summary of statistics for fields matched and unmatched in your account records with Data.com account records. This object is available in API version 37.0 and later.

[DataAssessmentValueMetric](#)

Summarizes the number of fields matched for your account records with Data.com account records. This object is available in API version 37.0 and later.

[DatacloudCompany](#)

Represents the fields for Data.com company records. This object is available in API version 30.0 or later.

[DatacloudContact](#)

The fields and properties for Data.com contact records. This object is available in API version 30.0 or later.

[DatacloudDandBCompany](#)

Represents a set of read-only fields that are used to return D&B company data from Data.com API calls. This object is available in API version 30.0 or later.

[DatacloudOwnedEntity](#)

Represents fields in the DatacloudOwnedEntity object. The DatacloudOwnedEntity object tracks user-purchased records. This object is available in API version 30.0 or later.

[DatacloudPurchaseUsage](#)

Represents an object used to identify and track Data.com record purchases. This object is available in API version 30.0 or later.

[DatacloudSocialHandle](#)

Returns normalized URLs with userids for different social media used by Data.com contacts. The DatacloudSocialHandle object is a child object of the DatacloudContact object. This object is available in API version 30.0 or later.

Standard Objects

[DatedConversionRate](#)

Represents the dated exchange rates used by an organization for which the multicurrency and the effective dated currency features are enabled.

[DcSocialProfile](#)

The `DcSocialProfile` object is a read-only object accessible only through the Data.com Social Key API. This object is available in API version 32.0 or later.

[DcSocialProfileHandle](#)

The `DcSocialProfileHandle` object, a child object to `DcSocialProfile`, is a read-only object accessible only through the Data.com Social Key API. The `DcSocialProfileHandle` object is a child object of the `DcSocialProfile` object. This object is available in API version 32.0 or later.

[DeclinedEventRelation](#)

Represents invitees with the status `Declined` for a given event. This object is available in API versions 29.0 and later.

[Division](#)

A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.

[DivisionLocalization](#)

When the Translation Workbench is enabled for your organization, the `DivisionLocalization` object provides the translation of the label for a division.

[Document](#)

Represents a file that a user has uploaded. Unlike Attachment records, documents are not attached to a parent object.

[DocumentAttachmentMap](#)

Maps the relationship between an `EmailTemplate` and its attachment, which is stored as a Document.

[DocumentTag](#)

Associates a word or short phrase with a Document.

[Domain](#)

Read-only object that represents a custom Web address assigned to a site in your organization. This object is available in API version 26.0 and later.

[DomainSite](#)

Read-only junction object that joins together the Site and Domain objects. This object is available in API version 26.0 and later.

[DuplicateRecordItem](#)

Represents an individual record that's part of a duplicate record set. Use this object to create custom report types.

[DuplicateRecordSet](#)

Represents a group of records that have been identified as duplicates. Each duplicate record set contains one or more duplicate record items. Use this object to create custom report types.

[DuplicateRule](#)

Represents a duplicate rule for detecting duplicate records.

[EmailDomainKey](#)

Represents a domain key for an organization's domain, used to authenticate outbound email that Salesforce sends on the organization's behalf. This object is available in API version 28.0 and later.

[EmailMessage](#)

Represents an email in Salesforce.

Standard Objects

[EmailMessageRelation](#)

Represents the relationship between an email and contacts, leads, and users. This object is available in API version 37.0 and later.

[EmailServicesAddress](#)

An email service address.

[EmailServicesFunction](#)

An email service.

[EmailStatus](#)

Represents the status of email sent.

[EmailTemplate](#)

Represents a template for mass email, or email sent when the activity history related list of a record is modified.

[Entitlement](#)

Represents the customer support an Account or Contact is eligible to receive. This object is available in API version 18.0 and later. Entitlements may be based on an Asset, Product2, or ServiceContract.

[EntitlementContact](#)

Represents a Contact eligible to receive customer support via an Entitlement. This object is available in API version 18.0 and later.

[EntitlementFeed](#)

Represents a single feed item in the feed displayed on the detail page for an entitlement. An entitlement feed shows recent changes to an entitlement record for any fields that are tracked in feeds, and comments and posts about the record. It's a useful way to stay up-to-date with changes to entitlements. This object is available in API version 23.0 and later.

[EntitlementHistory](#)

Represents the changes to field values on an Entitlement. This object is available in API version 18.0 and later.

[EntitlementTemplate](#)

Represents predefined terms of customer support for a product (Product2). This object is available in API version 18.0 and later.

[EntityHistory](#)

Represents historical information about an object's changed field values. This object is only available to users with the "View All Data" permission. This object is unavailable beginning with API version 8.0. Use the object-specific History objects instead: CaseHistory, ContractHistory, LeadHistory, OpportunityFieldHistory, OpportunityHistory ProcessInstanceHistory, QuantityForecastHistory, RevenueForecastHistory, or SolutionHistory.

[EntityMilestone](#)

Represents a required step in a customer support process on a record. The Salesforce user interface uses the term "object milestone." This object is available in API version 37.0 and later.

[EntitySubscription](#)

Represents a subscription for a user following a record or another user. This object is available in API version 34.0 and later.

[EnvironmentHubMember](#)

Represents a member organization in the Environment Hub. This object is available in API version 29.0 and later.

[Event](#)

Represents an event in the calendar. In the user interface, event and task records are collectively referred to as activities.

[EventFeed](#)

Represents a single feed item in the feed on an Event. This object is available in API version 20.0 and later.

[EventLogFile](#)

Represents event log files for event monitoring. This object is available in API versions 32.0 and later.

Standard Objects

[EventRelation](#)

Represents people (a user, a lead, or contacts) or a resource (such as a conference room) invited to an event. This object lets you add or remove invitees from an event and use the API to manage invitees' responses to invitations. If Shared Activities is enabled, this object doesn't support triggers, workflow, or data validation rules.

[EventTag](#)

Associates a word or short phrase with an Event.

[EventWhoRelation](#)

Represents the relationship between an event and a lead or contacts. This derived object is a filtered version of the [EventRelation](#) on page 948 object; that is, IsParent is `true` and IsWhat is `false`. It doesn't represent relationships to invitees or to accounts, opportunities, or other objects. This object is available in API versions 29.0 and later.

[ExternalDataSource](#)

Represents an external data source, which defines connection details for integration with data and content that are stored outside the Salesforce org. This object is available in API version 27.0 and later.

[ExternalDataUserAuth](#)

Stores authentication settings for a Salesforce user to access an external system. The external system must be defined in an external data source or a named credential that's configured to use per-user authentication. This object is available in API version 27.0 and later.

[FeedAttachment](#)

Represents an attachment to a feed item, such as a file attachment or a link. Use FeedAttachment to add various attachments to one feed item. This object is available in API version 36.0 and later.

[FeedComment](#)

Represents a comment added to a feed by a user. This object is available in API version 18.0 and later.

[FeedItem](#)

FeedItem represents an entry in the feed, such as changes in a record feed, including text posts, link posts, and content posts. This object is available in API version 21.0 and later. This object replaces FeedPost.

[FeedLike](#)

Indicates that a user has liked a feed item. This object is available in API version 21.0 and later.

[FeedPollChoice](#)

Shows the choices for a poll posted in the feed. This object is available in API version 29.0 and later.

[FeedPollVote](#)

Shows how users voted on a poll posted in the feed. This object is available in API version 29.0 and later.

[FeedPost](#)

FeedPost represents the following types of changes in a record feed, such as AccountFeed: text posts, link posts, and content posts. This object is available in API version 18.0 through 21.0. FeedPost is no longer available in later versions. Starting with API version 21.0, use FeedItem to represent text posts, link posts, and content posts in feeds.

[FeedRevision](#)

Holds the revision history of a specific feed item or comment, including a list of attributes that changed for each revision. This object is available in API version 34.0 and later.

[FeedTrackedChange](#)

Represents an individual field change or set of field changes. A FeedTrackedChange is a child object of a record feed, such as AccountFeed. This object is available in API version 18.0 and later..

Standard Objects

[FieldHistoryArchive](#)

Represents field history values for all objects that retain field history. This is a BigObject, available only to users with the "Retain Field History" permission. This object is available in API version 29.0 and later.

[FieldPermissions](#)

Represents the enabled field permissions for the parent PermissionSet. This object is available in API version 24.0 and later.

[FiscalYearSettings](#)

Settings to define a custom or standard fiscal year for your organization. This object has a parent-child relationship with the Period object.

[FlexQueueItem](#)

Represents an asynchronous Apex job in the Apex flex queue. Provides information about the job type and flex queue position of the AsyncApexJob. This object is available in API version 36.0 and later.

[FlowInterview](#)

Represents a flow interview. A *flow interview* is a running instance of a flow.

[FlowInterviewOwnerSharingRule](#)

Represents the rules for sharing a FlowInterview with users other than the owner. This object is available in API version 33.0 and later.

[FlowInterviewShare](#)

Represents a sharing entry on a FlowInterview. This object is available in API version 33.0 and later.

[Folder](#)

Represents a repository for a Document, EmailTemplate, Report, or Dashboard. Only one type of item can be contained in a folder.

[FolderedContentDocument](#)

Represents the relationship between a parent and child ContentFolderItem in a ContentWorkspace.

[ForecastingAdjustment](#)

This object represents an individual sales manager's adjustment for a subordinate's forecast via a ForecastingItem. Available in API versions 26 and greater. This object is separate from the ForecastingOwnerAdjustment object, which represents forecast users' adjustments of their *own* forecasts.

[ForecastingFact](#)

This is a read-only object linking a ForecastingItem with its opportunities, such as opportunities that share the same owner or forecast category and have a closing date within the period of the forecasting item. Available in API versions 26 and greater.

[ForecastingItem](#)

This is a read-only object used for individual forecast amounts. Users see amounts based on their perspectives and forecast roles. The amounts users see include one of the following when forecasting in revenue: `AmountWithoutAdjustments`, `AmountWithoutManagerAdjustment`, `ForecastAmount`, `OwnerOnlyAmount`. The amounts users see include one of the following when forecasting in quantity: `QuantityWithoutAdjustments`, `QuantityWithoutManagerAdjustment`, `ForecastQuantity`, `OwnerOnlyQuantity`. Available in API versions 26 and greater.

[ForecastingOwnerAdjustment](#)

This object represents an individual forecast user's adjustment of their *own* forecast via a ForecastingItem. Available in API versions 33 and greater. This object is separate from the ForecastingAdjustment object, which represents managers' adjustments of *subordinates'* forecasts.

Standard Objects

[ForecastingQuota](#)

This object represents an individual user's quota for a specified time period. The "Manage Quotas" user permission is required for creating, updating, or deleting quotas. (Users can only edit their subordinates' quotas, not their own.) The "View All Forecasts" permission is required to View any user's forecast, regardless of the forecast role hierarchy. Available in API versions 25 and greater. Forecast managers can view the forecasts of subordinates who report to them in the forecast hierarchy.

[ForecastingType](#)

This object is used to identify the forecast type associated with [ForecastingAdjustment](#), [ForecastingOwnerAdjustment](#), [ForecastingQuota](#), [ForecastingFact](#), and [ForecastingItem](#) objects. Available in API versions 30.0 and greater.

[ForecastShare](#)

Represents the sharing of a customizable forecast at a given role and territory.

[Goal](#)

The Goal object represents the components of a goal such as its name, description, and status.

[GoalFeed](#)

Represents a single feed item in the feed displayed for a Goal record. The goal feed shows changes to a goal for fields that are tracked in feeds, posts, and comments about the goal.

[GoalHistory](#)

This read-only object contains historical information about changes that have been made to the Goal object.

[GoalLink](#)

Represents the relationship between two goals. This is a many-to-many relationship, meaning that each goal can link to many other goals.

[GoalShare](#)

Represents a sharing entry on a Goal object.

[GoogleDoc](#)

Represents a link to a Google Document. This object is available in API version 14.0 and later.

[Group](#)

A set of User records.

[GroupMember](#)

Represents a User or Group that is a member of a public group.

[HashtagDefinition](#)

HashtagDefinition represents hashtag (#) topics in public Chatter posts and comments. Public posts and comments include those on profiles and in public groups, but not those on records or in private groups. This object is available in API version 26.0 and later.

[Holiday](#)

Represents a period of time during which your customer support team is unavailable. Business hours and escalation rules associated with business hours are suspended during any holidays with which they are affiliated.

[Idea](#)

Represents an idea on which users are allowed to comment and vote, for example, a suggestion for an enhancement to an existing product or process. This object is available in API version 12 and later.

[IdeaComment](#)

Represents a comment that a user has submitted in response to an idea.

Standard Objects

[IdeaReputation](#)

Represents a collection of statistics and scores derived from a user's activity within an Ideas zone or internal organization. This object is available in API version 28.0 and later.

[IdeaReputationLevel](#)

Represents a reputation level within an Ideas zone or internal organization and is used by the system to calculate reputation. You can create up to 25 levels per zone or internal organization. This object is available in API version 28.0 and later.

[IdeaTheme](#)

Represents an invitation to community members to submit ideas that are focused on a specific topic. This object is available in API version 26 and later.

[KnowledgeableUser](#)

Represents a user identified as knowledgeable about a specific topic, and ranks them relative to other knowledgeable users. This object is available in API version 31.0 and later.

[KnowledgeArticle](#)

Provides read-only access to an article and the ability to delete the master article. This object is available in API version 19.0 and later.

[KnowledgeArticleVersion](#)

Provides a global view of standard article fields across all article types depending on their version. This object is available in API version 18.0 and later.

[KnowledgeArticleVersionHistory](#)

Enables read-only access to the full history of an article. This object is available in API version 25.0 and later.

[KnowledgeArticleViewStat](#)

Provides statistics on the number of views for the specified article across all article types. This object is read-only and available in API version 20 and later.

[KnowledgeArticleVoteStat](#)

Provides the weighted rating for the specified article on a scale of 1 to 5 across all article types. This object is read-only and available in API version 20 and later.

[Lead](#)

Represents a prospect or potential Opportunity.

[LeadCleanInfo](#)

Stores the metadata Data.com Clean uses to determine a lead record's clean status. Helps you automate the cleaning or related processing of lead records.

[LeadFeed](#)

Represents a single feed item in the feed displayed on the detail page for a lead record. This object is available in API version 18.0 and later.

[LeadHistory](#)

Represents the history of changes to the values in the fields of a lead.

[LeadOwnerSharingRule](#)

Represents the rules for sharing a lead with users other than the owner.

[LeadShare](#)

Represents a sharing entry on a Lead.

[LeadStatus](#)

Represents the status of a Lead, such as Open, Qualified, or Converted.

Standard Objects

[LeadTag](#)

Associates a word or short phrase with a Lead.

[LimitAllocationPerApp](#)

Represents a connected app quota for an API limit. This object is available in API version 30.0 and later.

[LineitemOverride](#)

A forecast override of a line item on an Opportunity. This read-only object for customizable forecasting has a child-parent relationship with OpportunityOverride.

[LinkedArticle](#)

Represents a Knowledge article that is attached to a record. This object is available in API version 37.0 and later.

[LinkedArticleFeed](#)

Represents a single feed item on a linked Knowledge article attached to a record. This object is available in API version 37.0 and later.

[LinkedArticleHistory](#)

Represents the history of changes made to tracked fields on a linked article. This object is available in API version 37.0 and later.

[ListView](#)

Represents a list view. A list view specifies a set of records for an object, based on specific criteria. This object is available in API version 32.0 and later.

[ListViewChart](#)

Represents a graphical chart that's displayed on Salesforce1 list views. The chart aggregates data that is filtered based on the list view that's currently displayed. This object is available in API version 33.0 and later.

[ListViewChartInstance](#)

Retrieves metadata for all standard and custom charts for a given entity in context of a given list view. This object is available in API versions 34.0 and later.

[LiveAgentSession](#)

This object is automatically created for each Live Agent session and stores information about the session. This object is available in API versions 28.0 and later.

[LiveAgentSessionHistory](#)

This object is automatically created for each Live Agent session and stores information about changes made to the session. This object is available in API versions 28.0 and later.

[LiveAgentSessionOwnerSharingRule](#)

Represents the rules for sharing a Live Agent session record with users other than the record owner. This object is available in API version 28.0 and later.

[LiveAgentSessionShare](#)

This object is automatically created for each Live Agent session and stores information about the session. This object is available in API versions 28.0 and later.

[LiveChatBlockingRule](#)

Represents a rule for blocking chat visitors' IP addresses from starting new chats with agents. This object is available in API version 34.0 and later.

[LiveChatButton](#)

Represents a button that allows visitors to request chats with Live Agent users. This object is available in API version 24.0 and later.

[LiveChatButtonDeployment](#)

Associates a Live Agent automated chat invitation with a specific deployment. This object is available in API versions 28.0 and later.

Standard Objects

[LiveChatButtonSkill](#)

Represents all the skills available to a LiveChatButton except the one currently assigned. To retrieve the skill currently assigned, query LiveChatButton. This object is available in API version 25.0 and later.

[LiveChatDeployment](#)

Represents the general settings for deploying Live Agent on a website. This object is available in API version 24.0 and later.

[LiveChatSensitiveDataRule](#)

Represents a rule for masking or deleting data of a specified pattern. Written as a regular expression (regex). This object is available in API version 35.0 and later.

[LiveChatTranscript](#)

This object is automatically created for each Live Agent chat session and stores information about the session. This object is available in API version 24.0 and later.

[LiveChatTranscriptEvent](#)

Captures specific events that occur over the lifetime of a chat. This object is available in API version 24.0 and later.

[LiveChatTranscriptHistory](#)

Represents changes to field values on a LiveChatTranscript object. This object is available in API version 24.0 and later.

[LiveChatTranscriptOwnerSharingRule](#)

Represents the rules for sharing a Live Agent chat transcript record with users other than the record owner. This object is available in API version 29.0 and later.

[LiveChatTranscriptShare](#)

Represents a sharing entry on a LiveChatTranscript object. This object is available in API version 24.0 and later.

[LiveChatTranscriptSkill](#)

Represents a join between LiveChatTranscript and Skill. This object is available in API version 25.0 and later.

[LiveChatUserConfig](#)

Represents a setting that controls the console settings for Live Agent users. This object is available in API version 24.0 and later.

[LiveChatUserConfigProfile](#)

Represents a join between LiveChatUserConfig and Profile. This object is available in API version 24.0 and later.

[LiveChatUserConfigUser](#)

Represents a join between LiveChatUserConfig and User. This object is available in API version 24.0 and later.

[LiveChatVisitor](#)

Represents a website visitor who has started or tried to start a chat session. This object is available in API version 24.0 and later.

[LoginEvent](#)

Represents a trackable user login event in your org. This object is available in API version 36.0 and later.

[LoginGeo](#)

Represents the geographic location of the user's IP address for a login event. Due to the nature of geolocation technology, the accuracy of geolocation fields (for example, country, city, postal code) may vary. This object is available in API version 34.0 and later.

[LoginHistory](#)

Represents the login history for all successful and failed login attempts for organizations and enabled portals. This object is available in API version 21.0 and later.

[LoginIp](#)

Represents a validated IP address. This object is available in version 28.0 and later.

Standard Objects

[LookedUpFromActivity](#)

This read-only object is displayed as a related list on an activity record (an event or a task); the list contains records that have custom lookup relationships from the activity to another object. This object is not queryable.

[MacroInstruction](#)

Represents an instruction in a macro. An instruction can specify the object that the macro interacts with, the context or publisher that the macro works within, the operation or action that the macro performs, and the target of the macro's actions. It is a useful way to programmatically define instructions, instead of using the macro widget in the console.

[MailmergeTemplate](#)

Represents a mail merge template (a Microsoft Word document) used for performing mail merges for your organization.

[MatchingRule](#)

Represents a matching rule that is used to identify duplicate records. This object is available in API version 33.0 and later.

[MatchingRuleItem](#)

Represents criteria used by a matching rule to identify duplicate records. This object is available in API version 33.0 and later.

[MetadataPackage](#)

Represents a managed or unmanaged package that has been developed in the org you're logged in to.

[MetadataPackageVersion](#)

Represents a package version (managed or unmanaged) that has been uploaded from the org you're logged in to.

[Metric](#)

The Metric object represents the components of a goal metric such as its name, metric type, and current value.

[MetricDataLink](#)

The link between the metric and the data source, such as a report.

[MetricDataLinkHistory](#)

This read-only object contains historical information about changes that have been made to the MetricDataLink object.

[MetricFeed](#)

Represents a single feed item in the feed displayed on a Metric record.

[MetricHistory](#)

This read-only object contains historical information about changes that have been made to the Metric object.

[MetricsDataFile](#)

Represents a data file containing usage metrics on all installations of a managed package in a Salesforce instance. This object is available in API version 30.0 and later.

[MetricShare](#)

Represents a sharing entry on a Metric object.

[MilestoneType](#)

Represents a milestone (required step in a customer support process). This object is available in API version 18.0 and later.

[Name](#)

Non-queryable object that provides information about foreign key traversals when the foreign key has more than one parent.

[NamedCredential](#)

Represents a named credential, which specifies the URL of a callout endpoint and its required authentication parameters in one definition. A named credential can be specified as an endpoint to simplify the setup of authenticated callouts. This object is available in API version 33.0 and later.

Standard Objects

[NavigationLinkSet](#)

Represents the navigation menu in a community. A navigation menu consists of items that users can click to go to other parts of the community. This object is available in API version 36.0 and later.

[NavigationMenuItem](#)

Represents a single menu item in a NavigationLinkSet. Use this object to create, delete, or update menu items in your community's navigation menu. This object is available in API version 36.0 and later.

[NavigationMenuItemLocalization](#)

Represents the translated value of a navigation menu item in a community. This object is available in API version 36.0 and later.

[Network](#)

Represents a community, which is a customizable public or private space where employees, end-customers, and partners can collaborate on best practices and business processes. Communities give you the opportunity to share information, records, and files with coworkers and related external stakeholders all in one place. This object is available in API version 26.0 and later.

[NetworkActivityAudit](#)

Represents an audit trail of moderation actions in Communities. This object is available in API version 30.0 and later.

[NetworkMember](#)

Represents a member of a community. Members can be either users in your company or external users with portal profiles. This object is available in API version 26.0 and later.

[NetworkMemberGroup](#)

Represents a group of members in a community. Members can be either users in your internal organization or external users assigned portal profiles. An administrator adds members to a community by adding a profile or a permission set, and any user with the profile or permission set becomes a member of the community. This object is available in API version 26.0 and later.

[NetworkModeration](#)

Represents a flag on an item in a community. This object is available in API version 30.0 and later.

[NetworkPageOverride](#)

Represents information about custom pages used to override the default pages in communities. You can create Community Builder (Site.com Studio) or Visualforce pages and override the default pages in a community. Using custom pages allows you to create a more personalized experience for your users. This object is available in API version 34.0 and later.

[NetworkSelfRegistration](#)

Represents the account that self-registering community members are associated with by default. Self-registering users in a community are required to be associated with an account, which the administrator must specify while setting up self-registration for the community. If an account isn't specified, Salesforce creates person accounts (when enabled) for self-registering users. This object is available in API version 34.0 and later.

[NewsFeed](#)

Represents a single feed item on a user's home tab. A Chatter feed shows recent changes to records that the user is following.

[Note](#)

Represents a note, which is text associated with a custom object or a standard object, such as a Contact, Contract, or Opportunity.

[OAuthToken](#)

Represents an OAuth access token for connected app authentication and can be used to create a user interface for token management. This object is available in API version 32.0 and later.

[NoteAndAttachment](#)

This read-only object contains all notes and attachments associated with an object.

[NoteTag](#)

Associates a word or short phrase with a Note.

Standard Objects

[ObjectPermissions](#)

Represents the enabled object permissions for the parent PermissionSet. This object is available in API version 24.0 and later.

[ObjectTerritory2AssignmentRule](#)

Represents a territory assignment rule that's associated with an object, such as Account. ObjectTerritory2AssignmentRuleItem can only be created or deleted if the BooleanFilter field on its corresponding ObjectTerritory2AssignmentRule is null. Available only if Enterprise Territory Management has been enabled for your organization.

[ObjectTerritory2AssignmentRuleItem](#)

A single row of selection criteria for an ObjectTerritory2AssignmentRule object. ObjectTerritory2AssignmentRuleItem can only be created or deleted if the BooleanFilter field on its corresponding ObjectTerritory2AssignmentRule object is a null value. Available only if Enterprise Territory Management has been enabled for your organization.

[ObjectTerritory2Association](#)

Represents an association (by assignment) between a territory and an object record, such as an account. Available only if Enterprise Territory Management has been enabled for your Salesforce org.

[OpenActivity](#)

This read-only object is displayed in a related list of open activities—future events and open tasks—related to an object. It includes activities for all contacts related to the object. OpenActivity fields for phone calls are only available if your organization uses Salesforce CRM Call Center.

[OperatingHours](#)

Represents the hours in which a service territory, service resource, or account is available for field service work. This object is available in API version 38.0 and later.

[OperatingHoursFeed](#)

Represents a single feed item on an operating hours record detail page. This object is available in API version 38.0 and later.

[OperatingHoursHistory](#)

Represents the history of changes made to tracked fields on an operating hours record. This object is available in API version 38.0 and later.

[Opportunity](#)

Represents an opportunity, which is a sale or pending deal.

[OpportunityCompetitor](#)

Represents a competitor on an Opportunity.

[OpportunityContactRole](#)

Represents the role that a Contact plays on an Opportunity.

[OpportunityFeed](#)

Represents a single feed item in the feed displayed on the detail page for an opportunity record. This object is available in API version 18.0 and later.

[OpportunityFieldHistory](#)

Represents the history of changes to the values in the fields of an opportunity. This object is available in versions 13.0 and later.

[OpportunityHistory](#)

Represents the stage history of an Opportunity.

[OpportunityLineItem](#)

Represents an opportunity line item, which is a member of the list of Product2 products associated with an Opportunity.

[OpportunityLineItemSchedule](#)

Represents information about the quantity, revenue distribution, and delivery dates for a particular OpportunityLineItem.

Standard Objects

[OpportunityOverride](#)

Represents a forecast override of an Opportunity. This read-only object is specific to customizable forecasting. It has a parent-child relationship with LineitemOverride.

[OpportunityOwnerSharingRule](#)

Represents a rule for sharing an opportunity with users other than the owner.

[OpportunityPartner](#)

This read-only object represents a partner relationship between an Account and an Opportunity. This object is automatically created when a Partner object is created for a partner relationship between an account and an opportunity.

[OpportunityShare](#)

Represents a sharing entry on an Opportunity.

[OpportunitySplit](#)

OpportunitySplit credits one or more opportunity team members with a portion of the opportunity amount. This object is available in API version 16.0 and later for pilot customers, and version 28.0 and later for others.

[OpportunitySplitType](#)

OpportunitySplitType provides unique labels and behavior for each split type. This object is available in API version 28.0 and later.

[OpportunityStage](#)

Represents the stage of an Opportunity in the sales pipeline, such as New Lead, Negotiating, Pending, Closed, and so on.

[OpportunityTag](#)

Associates a word or short phrase with an Opportunity.

[OpportunityTeamMember](#)

Represents a User on the opportunity team of an Opportunity.

[Order](#)

Represents an order associated with a contract or an account.

[OrderFeed](#)

Represents a single feed item in the feed displayed on an order.

[OrderHistory](#)

Represents historical information about changes that have been made to the standard fields of the associated order, or to any custom fields with history tracking enabled.

[OrderItem](#)

Represents an order product that your organization sells.

[OrderItemFeed](#)

Represents a single feed item in the feed displayed on the detail page for an order product record.

[OrderItemHistory](#)

Represents the history of changes to the values in the fields of an order product.

[OrderOwnerSharingRule](#)

Represents a rule which determines order sharing access for the order's owners.

[Organization](#)

Represents key configuration information for an organization.

[OrgWideEmailAddress](#)

Represents an organization-wide email address for user profiles.

Standard Objects

[OwnedContentDocument](#)

Represents a file owned by a user. This object is available in version 30.0 and later.

[PackageLicense](#)

Represents a license for an installed managed package. This object is available in API version 31.0 and later.

[PackagePushError](#)

Represents an error encountered during a push request. The number of PackagePushError records created depends on the number of push jobs in the request that result in an error.

[PackagePushJob](#)

Represents an individual push job for upgrading a package in an org from one version to another version. There can be multiple push jobs created for one push request. For example, if you want to upgrade five orgs as part of one push, you have one PackagePushRequest record and five PackagePushJob records.

[PackagePushRequest](#)

Represents the push request for upgrading a package in one or many orgs from one version to another version.

[PackageSubscriber](#)

Represents an installation of a package in an org. This object contains installation information for managed packages developed in the org you're logged in to.

[Partner](#)

Represents a partner relationship between two Account records or between an Opportunity and an Account.

[PartnerNetworkConnection](#)

Represents a Salesforce to Salesforce connection between Salesforce organizations.

[PartnerNetworkRecordConnection](#)

Represents a record shared between Salesforce organizations using Salesforce to Salesforce.

[PartnerNetworkSyncLog](#)

Represents the Org Sync Log tab in Salesforce, where Salesforce administrators can track the replication of record inserts and updates being performed in Organization Sync. The Connection Detail page for the replication connection also displays the Org Sync Log's twenty most recent entries, and provides a link to the log.

[PartnerRole](#)

Represents a role for an account Partner, such as consultant, supplier, and so on.

[Period](#)

Represents a fiscal period defined in FiscalYearSettings.

[PermissionSet](#)

Represents a set of permissions that's used to grant additional access to one or more users without changing their profile or reassigning profiles. This object is available in API version 22.0 and later.

[PermissionSetAssignment](#)

Represents the association between a User and a PermissionSet. This object is available in API version 22.0 and later.

[PermissionSetLicense](#)

Represents a license that's used to enable one or more users to receive a specified permission without changing their profile or reassigning profiles. You can use permission set licenses to grant access, but not to deny access. This object is available in API version 29.0 and later.

[PermissionSetLicenseAssign](#)

Represents the association between a User and a PermissionSetLicense. This object is available in API version 29.0 and later.

Standard Objects

[PlatformAction](#)

PlatformAction is a virtual read-only object. It enables you to query for actions displayed in the UI, given a user, a context, device format, and a record ID. Examples include standard and custom buttons, quick actions, and productivity actions.

[PresenceUserConfig](#)

Represents a configuration that determines a presence user's settings. This object is available in API version 32.0 and later.

[PresenceUserConfigProfile](#)

Represents a configuration that determines the settings that are assigned to presence users who are assigned to a specific profile. User-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

[PresenceUserConfigUser](#)

Represents a configuration that determines the settings that are assigned to a presence user. These user-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

[Pricebook2](#)

Represents a price book that contains the list of products that your org sells.

[Pricebook2History](#)

Represents historical information about changes that have been made to the standard fields of the associated Pricebook2, or to any custom fields with history tracking enabled. This object is available in API version 38.0 and later.

[PricebookEntry](#)

Represents a product entry (an association between a Pricebook2 and Product2) in a price book.

[ProcessDefinition](#)

Represents the definition of a single approval process.

[ProcessInstance](#)

Represents an instance of a single, end-to-end approval process. Use this and the node, step, and workitem process instance objects to create approval history reports.

[ProcessInstanceHistory](#)

This read-only object shows all steps and pending approval requests associated with an approval process (ProcessInstance).

[ProcessInstanceStep](#)

Represents one work item in an approval process (ProcessInstance).

[ProcessInstanceNode](#)

Represents a step in an instance of an approval process. Compare to ProcessNode, which describes the step in a process definition. Use this object to retrieve approval history.

[ProcessInstanceWorkitem](#)

Represents a user's pending approval request.

[ProcessNode](#)

Describes a step in a process definition. Compare to ProcessInstanceNode, which describes the step in a running process.

[Product2](#)

Represents a product that your org sells.

[Product2Feed](#)

Represents a single feed item in the feed displayed on the detail page for a product record. This object is available in API version 18.0 and later.

[ProductEntitlementTemplate](#)

Represents predefined terms of customer support (Entitlement) that users can add to products (Product2).

Standard Objects

[Profile](#)

Represents a profile, which defines a set of permissions to perform different operations, such as querying, adding, updating, or deleting information.

[ProfileSkill](#)

Represents a profile skill, which describes a user's professional knowledge. This is a global record for the organization, and users are associated through the ProfileSkillUser object.

[ProfileSkillEndorsement](#)

Represents a detail relationship of ProfileSkillUser. An endorsement of a profile skill shows approval and support of another user's publicly declared skill.

[ProfileSkillEndorsementFeed](#)

Represents a single feed item in the feed displayed on the detail page for a ProfileSkillEndorsement record. This object is available in API version 34.0 and later.

[ProfileSkillEndorsementHistory](#)

Represents the history of changes to the fields of a ProfileSkillEndorsement.

[ProfileSkillFeed](#)

Represents a single feed item in the feed displayed on the detail page for a profile skill.

[ProfileSkillHistory](#)

Represents the history of changes to the fields of a ProfileSkill.

[ProfileSkillShare](#)

Represents a sharing entry on a ProfileSkill.

[ProfileSkillUser](#)

Represents a detail relationship of User. The object connects profile skills with users.

[ProfileSkillUserFeed](#)

Represents a single feed item in the feed displayed on the detail page for a ProfileSkillUser record. This object is available in API version 34.0 and later.

[ProfileSkillUserHistory](#)

Represents the history of changes to the fields of a ProfileSkillUser.

[PushTopic](#)

[QuantityForecast](#)

Represents a quantity-based forecast.

[QuantityForecastHistory](#)

Represents historical information about quantity-based forecasts that have been submitted (saved) in the user interface.

[QueueRoutingConfig](#)

Represents the settings that determine how work items are routed to agents. This object is available in API version 32.0 and later.

[Question](#)

Represents a question in a community that users can view and reply to.

[QuestionDataCategorySelection](#)

A data category selection represents a data category that classifies a question.

[QuestionReportAbuse](#)

Represents a user-reported abuse on a Question in a Chatter Answers community. This object is available in API version 24.0 and later.

Standard Objects

[QuestionSubscription](#)

Represents a subscription for a user following a Question. This object is available in API version 24.0 and later.

[QueueSobject](#)

Represents the mapping between a queue Group and the sObject types associated with the queue, including custom objects.

[QuickText](#)

This object stores a snippet of text that allows an agent to send a quick response to a customer in the Live Agent console. This object is available in API version 24.0 and later.

[QuickTextHistory](#)

Represents changes to field values on a QuickText object. This object is available in API version 24.0 and later.

[QuickTextOwnerSharingRule](#)

Represents a rule for sharing a QuickText object with users other than the owner.

[QuickTextShare](#)

Represents a sharing entry on a QuickText object. This object is available in API version 24.0 and later.

[Quote](#)

The Quote object represents a quote, which is a record showing proposed prices for products and services. Available in API version 18.0 and later.

[QuoteDocument](#)

Represents a quote in document format. Available in API version 18.0 and later.

[QuoteLineItem](#)

The QuoteLineItem object represents a quote line item, which is a member of the list of Product2 products associated with a Quote, along with other information about those line items on that quote. Available in API version 18.0 and later.

[RecentlyViewed](#)

Represents records that the current user has recently viewed or referenced (by viewing a related record).

[RecordType](#)

Represents a record type.

[RecordTypeLocalization](#)

Represents the translated value of a label for a record type when the Translation Workbench is enabled for your organization.

[Reply](#)

Represents a reply that a user has submitted to a question in an answers community.

[ReplyReportAbuse](#)

Represents a user-reported abuse on a Reply in a Chatter Answers community. This object is available in API version 24.0 and later.

[Report](#)

Represents a report, a set of data that meets certain criteria, displayed in an organized way. Access is read-only. This object is available in API version 20.0 and later.

[ReportFeed](#)

Represents a single feed item in the feed displayed on a report. This object is available in API version 20.0 and later.

[ReportTag](#)

Associates a word or short phrase with a Report. This object is available in API version 20.0 and later.

[ReputationLevel](#)

Represents a reputation level defined for a community. This object is available in API version 32.0 and later.

Standard Objects

[ReputationLevelLocalization](#)

Represents the translated value of a reputation level. Reputation level localization only applies for reputation levels in communities. This object is available in API version 35.0 and later.

[ReputationPointsRule](#)

Represents the reputation point rules for a community. Each rule specifies an action that community members can earn points from and the points associated with those actions in a particular community. This object is available in API version 32.0 and later.

[ResourceAbsence](#)

Represents a time period in which a service resource is unavailable to work. This object is available in API version 38.0 and later.

[ResourceAbsenceFeed](#)

Represents a single feed item on a resource absence record detail page. This object is available in API version 38.0 and later.

[ResourceAbsenceHistory](#)

Represents the history of changes made to tracked fields on a resource absence. This object is available in API version 38.0 and later.

[RevenueForecast](#)

Represents a revenue-based forecast.

[RevenueForecastHistory](#)

Represents historical information about revenue-based forecasts that have been submitted (saved) in the user interface.

[RuleTerritory2Association](#)

Represents a record-assignment rule and its association to an object, such as Account. Available only if Enterprise Territory Management has been enabled for your organization.

[SamlSsoConfig](#)

Represents a SAML Single Sign-On configuration. This object is available in API version 32.0 and later.

[Scontrol](#)

A custom s-control, which is custom content that is hosted by the system but executed by the client application.

[ScontrolLocalization](#)

The translated value of the field label for an s-control.

[SearchPromotionRule](#)

Represents a promoted search term, which is one or more keywords that you associate with a Salesforce Knowledge article. When a user's search query includes these keywords, the associated article is returned first in search results. This object is available in API version 31.0 and later.

[SecureAgent](#)

Represents a Secure Agent that connects Salesforce to on-premises external data sources like SharePoint 2010 and 2013. This object is available in API version 32.0 and later.

[SecureAgentsCluster](#)

Represents a cluster consisting of several Secure Agents on different servers. Clusters provide failover protection if an agent on a particular server becomes inaccessible. This object is available in API version 35.0 and later.

[SelfServiceUser](#)

Represents a Contact who has been enabled to use your organization's Self-Service portal, where he or she can obtain online support.

[ServiceAppointment](#)

Represents an appointment to complete field service work for a customer. This object is available in API version 38.0 and later.

[ServiceAppointmentFeed](#)

Represents a single feed item on a service appointment record detail page. This object is available in API version 38.0 and later.

Standard Objects

[ServiceAppointmentHistory](#)

Represents the history of changes made to tracked fields on a service appointment. This object is available in API version 38.0 and later.

[ServiceAppointmentOwnerSharingRule](#)

Represents the rules for sharing a service appointment with user records other than the owner or anyone above the owner in the role hierarchy. This object is available in API version 38.0 and later.

[ServiceAppointmentShare](#)

Represents a sharing entry on a service appointment. This object is available in API version 38.0 and later.

[ServiceChannel](#)

Represents a channel of work items that are received from your organization—for example, cases, chats, or leads. This object is available in API version 32.0 and later.

[ServiceChannelStatus](#)

Represents the status that's associated with a specific service channel. This object is available in API version 32.0 and later.

[ServiceContract](#)

Represents a customer support contract (business agreement). This object is available in API version 18.0 and later.

[ServiceContractFeed](#)

Represents a single feed item in the feed displayed on the detail page for a service contract record. This object is available in API version 23.0 and later.

[ServiceContractHistory](#)

Represents the history of changes to the values in the fields on a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

[ServiceContractOwnerSharingRule](#)

Represents the rules for sharing a ServiceContract (customer service agreement) with users other than the owner. This object is available in API version 18.0 and later.

[ServiceContractShare](#)

Represents a sharing entry on a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

[ServicePresenceStatus](#)

Represents a presence status that can be assigned to a service channel. This object is available in API version 32.0 and later.

[ServiceResource](#)

Represents a service technician or dispatcher in field service. This object is available in API version 38.0 and later.

[ServiceResourceCapacity](#)

Represents the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period. This object is available in API version 38.0 and later.

[ServiceResourceCapacityFeed](#)

Represents a single feed item on a service resource capacity record detail page. This object is available in API version 38.0 and later.

[ServiceResourceCapacityHistory](#)

Represents the history of changes made to tracked fields on a service resource capacity record. This object is available in API version 38.0 and later.

[ServiceResourceFeed](#)

Represents a single feed item on a service resource record detail page. This object is available in API version 38.0 and later.

[ServiceResourceHistory](#)

Represents the history of changes made to tracked fields on a service resource. This object is available in API version 38.0 and later.

Standard Objects

[ServiceResourceOwnerSharingRule](#)

Represents the rules for sharing a service resource with user records other than the owner or anyone above the owner in the role hierarchy. This object is available in API version 38.0 and later.

[ServiceResourceShare](#)

Represents a sharing entry on a service resource. This object is available in API version 38.0 and later.

[ServiceResourceSkill](#)

Represents a skill that a service resource possesses. This object is available in API version 38.0 and later.

[ServiceResourceSkillFeed](#)

Represents a single feed item on a service resource skill record detail page. This object is available in API version 38.0 and later.

[ServiceResourceSkillHistory](#)

Represents the history of changes made to tracked fields on a service resource skill. This object is available in API version 38.0 and later.

[ServiceTerritory](#)

Represents a region in which field service work can be performed. This object is available in API version 38.0 and later.

[ServiceTerritoryFeed](#)

Represents a single feed item on a service territory record detail page. This object is available in API version 38.0 and later.

[ServiceTerritoryHistory](#)

Represents the history of changes made to tracked fields on a service territory. This object is available in API version 38.0 and later.

[ServiceTerritoryMember](#)

Represents a service resource who is able to be assigned to service appointments in the service territory. This object is available in API version 38.0 and later.

[ServiceTerritoryMemberFeed](#)

Represents a single feed item on a service territory member record detail page. This object is available in API version 38.0 and later.

[ServiceTerritoryMemberHistory](#)

Represents the history of changes made to tracked fields on a service territory member. This object is available in API version 38.0 and later.

[SessionPermSetActivation](#)

The SessionPermSetActivation object represents a permission set assignment activated during an individual user session. When a SessionPermSetActivation object is inserted into a permission set, an activation event fires, allowing the permission settings to apply to the user's specific session. This object is available in API versions 37.0 and later as part of the Session-Based Permission Sets Developer Preview.

[SetupAuditTrail](#)

Represents changes you or other admins made in your org's Setup area for at least the last 180 days. This object is available in API version 15.0 and later.

[SetupEntityAccess](#)

Represents the enabled setup entity access settings (such as for Apex classes) for the parent PermissionSet. This object is available in API version 25.0 and later.

[SignupRequest](#)

Represents a request for a new Trialforce sign-up. This object is available in API version 27.0 and later.

[Site](#)

Represents a public website that is integrated with an Organization. This object is generally available in API version 16.0 and later.

Standard Objects

[SiteDomain](#)

SiteDomain is a read-only object, and a one-to-many replacement for the Site.TopLevelDomain field. This object is available in API version 21.0, and has been deprecated as of API version 26.0. In API version 26.0 and later, use the [Domain](#) and [DomainSite](#) objects instead.

[SiteHistory](#)

Represents the history of changes to the values in the fields of a site. This object is generally available in API version 18.0 and later.

[Skill](#)

Represents a category or group that Live Agent users or field service resources can be assigned to. This object is available in API version 24.0 and later.

[SkillProfile](#)

Represents a join between Skill and Profile. This object is available in API version 24.0 and later.

[SkillRequirement](#)

Represents a skill that is required to complete a particular task. Skill requirements can be added to work types, work orders, and work order line items. This object is available in API version 38.0 and later.

[SkillRequirementFeed](#)

Represents a single feed item on a skill requirement record detail page. This object is available in API version 38.0 and later.

[SkillRequirementHistory](#)

Represents the history of changes made to tracked fields on a skill requirement. This object is available in API version 38.0 and later.

[SkillUser](#)

Represents a join between Skill and User. This object is available in API version 24.0 and later.

[SlaProcess](#)

Represents an entitlement process associated with an Entitlement. This object is available in API version 19.0 and later.

[Solution](#)

Represents a detailed description of a customer issue and the resolution of that issue.

[SolutionFeed](#)

Represents a single feed item in the feed displayed on the detail page for a solution record. This object is available in API version 18.0 and later.

[SolutionHistory](#)

Represents the history of changes to the values in the fields of a solution.

[SolutionStatus](#)

Represents the status of a Solution, such as Draft, Reviewed, and so on.

[SolutionTag](#)

Associates a word or short phrase with a Solution.

[SOSDeployment](#)

Represents the general settings for deploying SOS video call capability in a native mobile application. This object is available in API version 34.0 and later.

[SOSSession](#)

This object is automatically created for each SOS session and stores information about the session. This object is available in API versions 34.0 and later.

[SOSSessionActivity](#)

Captures information about specific events that occur during an SOS video call, such as when an SOS call begins or ends. This object is available in API version 34.0 and later.

Standard Objects

[SOSSessionHistory](#)

This object is automatically created for each SOS session and stores information about changes made to the session. This object is available in API versions 34.0 and later.

[SOSSessionOwnerSharingRule](#)

Represents the rules for sharing an SOS session record with users other than the record owner. This object is available in API version 34.0 and later.

[SOSSessionShare](#)

Represents a sharing entry on an SOS session. This object is available in API version 34.0 and later.

[StaticResource](#)

Represents a static resource that can be used in Visualforce markup.

[StreamingChannel](#)

[TagDefinition](#)

Defines the attributes of child Tag objects.

[Task](#)

Represents a business activity such as making a phone call or other to-do items. In the user interface, Task and Event records are collectively referred to as activities.

[TaskFeed](#)

Represents a single feed item in the feed on a Task. This object is available in API version 20.0 and later.

[TaskPriority](#)

Represents the importance or urgency of a Task, such as High, Normal, or Low.

[TaskRelation](#)

Represents the relationship between a task and a lead, contacts, and other objects related to the task. If Shared Activities is enabled, this object doesn't support triggers, workflow, or data validation rules. This object is available in API version 24.0 and later.

[TaskStatus](#)

Represents the status of a Task, such as Not Started, Completed, or Closed.

[TaskTag](#)

Associates a word or short phrase with a Task.

[TaskWhoRelation](#)

Represents the relationship between a task and a lead or contacts. This object is available in API version 29.0 and later.

[TenantSecret](#)

This object stores an encrypted organization-specific key fragment that is used with the master secret to produce organization-specific data encryption keys. This object is available in API version 34.0 and later.

[Territory](#)

Represents a flexible collection of accounts and users where the users have at least read access to the accounts, regardless of who owns the accounts. Only available if territory management has been enabled for your organization.

[Territory2](#)

Represents a sales territory. Available only if Enterprise Territory Management has been enabled for your organization.

[Territory2Model](#)

Represents a territory model. Available only if Enterprise Territory Management has been enabled for your organization.

[Territory2ModelHistory](#)

Represents the history of changes to the values in the fields on a territory model. Available only if Enterprise Territory Management has been enabled for your organization.

Standard Objects

[Territory2Type](#)

Represents a category for territories (Territory2). Every Territory2 must have a Territory2Type. Available only if Enterprise Territory Management has been enabled for your organization.

[TestSuiteMembership](#)

Associates an Apex class with an ApexTestSuite. This object is available in API version 36.0 and later.

[ThirdPartyAccountLink](#)

Represents the list of external users who authenticated using an Auth. Provider. This object is available in API version 32.0 and later.

[TimeSlot](#)

Represents a period of time on a specified day of the week during which field service work can be performed. Operating hours consist of one or more time slots. This object is available in API version 38.0 and later.

[TimeSlotHistory](#)

Represents the history of changes made to tracked fields on a time slot. This object is available in API version 38.0 and later.

[Topic](#)

Represents a topic on a Chatter post or record. This object is available in API version 28.0 and later.

[TopicAssignment](#)

Represents the assignment of a topic to a specific feed item, record, or file. This object is available in API version 28.0 and later.

[TopicFeed](#)

Represents a single feed item on a topic page. This object is available in API version 29.0 and later.

[TopicLocalization—Beta](#)

Represents the translated version of a topic name. Topic localization applies only to navigational and featured topics in communities. This object is available in API version 33.0 and later.

[TwoFactorInfo](#)

Stores a user's secret for two-factor operations. Use this object when customizing two-factor authentication in your organization. This object is available in API version 32.0 and later.

[TwoFactorMethodsInfo](#)

Stores information about which identity verification methods a user has registered. This object is available in API version 37.0 and later.

[TwoFactorTempCode](#)

Stores information about a user's temporary identity verification code. This object is available in API version 37.0 and later.

[UndecidedEventRelation](#)

Represents invitees with the status Not Responded for a given event. This object is available in API versions 29.0 and later.

[User](#)

Represents a user in your organization.

[UserAccountTeamMember](#)

Represents a User on the default account team of another User.

[UserAppInfo](#)

Stores the last Lightning app logged in to. This object is available in API version 38.0 and later.

[UserAppMenuCustomization](#)

Represents an individual user's settings for items in the Force.com app menu or App Launcher. This object is available in API version 35.0 and later.

Standard Objects

[UserAppMenuItemCustomizationShare](#)

Represents a sharing entry on a UserAppMenuItemCustomization record. This object is available in API version 35.0 and later.

[UserAppMenuItem](#)

Represents the organization-wide settings for items in the Force.com app menu or App Launcher that the requesting user has access to in Setup. This object is available in API version 35.0 and later.

[UserConfigTransferButton](#)

Represents the association between a Live Agent configuration and a live chat button. This association allows users associated with a specific configuration to transfer chats to a button queue.

[UserConfigTransferSkill](#)

Represents the association between a Live Agent configuration and a skill. This association allows users associated with a specific configuration to transfer chats to agents who have that skill.

[UserCustomBadge](#)

Represents a custom badge for a user. This object is available in API version 38.0 and later.

[UserCustomBadgeLocalization](#)

Represents the translated version of a custom badge for a user. This object is available in API version 38.0 and later.

[UserFeed](#)

Represents a single feed item in the feed displayed on a Chatter user profile feed. A user profile feed shows changes to a user record for fields that are tracked in feeds, and posts and comments about the record. This object is available in API version 18.0 and later.

[UserLicense](#)

Represents a user license in your organization. A user license entitles a user to specific functionality and determines the profiles and permission sets available to the user.

[UserLogin](#)

Represents the settings that affect a user's ability to log into an organization. To access this object, you need the `UserPermissions.ManageUsers` permission. This object is available in API version 29.0 and later.

[UserMembershipSharingRule](#)

Represents the rules for sharing user records from a source group to a target group. A user record contains details about a user. Users who are members of the source group can be shared with members of the target group. The source and target groups can be based on roles, portal roles, public groups, or territories. This object is available in API version 26.0 and later.

[UserPackageLicense](#)

Represents a license for an installed managed package, assigned to a specific user. This object is available in API version 31.0 and later.

[UserPreference](#)

Represents a functional preference for a specific user in your organization.

[UserProfile](#)

Represents a Chatter user profile.

[UserProfileFeed](#)

Represents a user profile feed, which tracks all actions by a user on records that can be tracked in a feed. This feed is displayed on the user profile page.

[UserProvAccount](#)

Represents information that links a Salesforce user account with an account in a third-party (target) system, such as Google, for users of connected apps with Salesforce user provisioning enabled. This object is available in API version 33.0 and later.

Standard Objects

[UserProvAccountStaging](#)

Temporarily stores user account information while a user completes the User Provisioning Wizard. This information that is stored in the UserProvAccount object when you click the button to collect and analyze accounts on the target system.

[UserProvMockTarget](#)

Represents an entity for testing user data before committing the data to a third-party system for user provisioning.

[UserProvisioningConfig](#)

Represents information for a flow to use during a user provisioning request process, such as the attributes for an update. This object is available in API version 34.0 and later.

[UserProvisioningLog](#)

Represents messages generated during the process of provisioning users for third-party applications. This object is available in API version 33.0 and later.

[UserProvisioningRequest](#)

Represents an individual provisioning request to create, update, or delete a single user account in a third-party service system (or another Salesforce organization). This object is available in API version 33.0 and later.

[UserProvisioningRequestOwnerSharingRule](#)

Represents a rule for sharing a UserProvisioningRequest object with users other than the owner. This object is available in API version 34.0 and later.

[UserProvisioningRequestShare](#)

Represents a sharing entry on a UserProvisioningRequest record. This object is available in API version 34.0 and later.

[UserRecordAccess](#)

Represents a user's access to a set of records. This object is read only and is available in API version 24.0 and later.

[UserRole](#)

Represents a user role in your organization.

[UserServicePresence](#)

Represents a presence user's real-time presence status. This object is available in API version 32.0 and later.

[UserShare](#)

Represents a sharing entry on a user record. This object is available in API version 26.0 and later.

[UserTeamMember](#)

Represents a single User on the default opportunity team of another User.

[UserTerritory](#)

Represents a User who has been assigned to a Territory.

[UserTerritory2Association](#)

Represents an association (by assignment) between a territory and a user record. Available only if Enterprise Territory Management has been enabled for your organization.

[VerificationHistory](#)

Represents the past six months of your org users' attempts to verify their identity. This object is available in API version 36.0 and later.

[Vote](#)

Represents a vote that a user has made on an Idea or a Reply.

[WebLink](#)

Represents a custom link to a URL or Scontrol.

Standard Objects

[WebLinkLocalization](#)

Represents the translated value of the field label for a custom link to a URL or s-control when the Translation Workbench is enabled for your organization.

[WorkAccess](#)

Used to grant or restrict user access to give badge definitions. Each badge definition record must have one WorkAccess record.

[WorkAccessShare](#)

Used to control Givers of WorkBadgeDefinition records.

[WorkBadge](#)

Represents information about who the badge was given to and which badge was given. A WorkBadge record is created for each recipient of a WorkBadgeDefinition.

[WorkBadgeDefinition](#)

Represents the attributes of a badge including the badge name, description, and image. Each WorkBadge record must have a lookup to a WorkBadgeDefinition since badge attributes (like badge name) are derived from the WorkBadgeDefinition object.

[WorkBadgeDefinitionHistory](#)

Represents the history of changes to the values in the fields of a WorkBadgeDefinition object.

[WorkBadgeDefinitionShare](#)

Represents a sharing entry on a WorkBadgeDefinition object.

[WorkCoaching](#)

Represents a single coaching relationship between two users. One of the users is defined as the coach and the other is defined as a coachee. WorkCoaching is feed-enabled so there is a private feed available to the coach and coachee.

[WorkCoachingFeed](#)

Represents a single feed item in the feed on the detail page for a coaching record.

[WorkCoachingHistory](#)

Represents the history of changes to the values in the fields of a WorkCoaching object.

[WorkCoachingShare](#)

Represents a sharing entry on a WorkCoaching object.

[WorkFeedback](#)

Represents the answer to a question that a person was asked via a feedback request. Also used to store offered feedback without linking it to a particular question.

[WorkFeedbackHistory](#)

Represents the history of changes to the values in the fields of a WorkFeedback object. Access is read-only.

[WorkFeedbackQuestion](#)

Represents a free-form text type or multiple choice question within a set of questions.

[WorkFeedbackQuestionHistory](#)

Represents the history of changes to the values in the fields of a WorkFeedbackQuestion.

[WorkFeedbackQuestionSet](#)

Represents a set of questions being asked. The question set is used to link all the individual requests where different recipients were asked the same set of questions on the same subject.

[WorkFeedbackQuestionSetHistory](#)

Represents the history of changes to the values in the fields of a WorkFeedbackQuestionSet object. Access is read-only.

Standard Objects

[WorkFeedbackQuestionSetShare](#)

Represents a sharing entry on a WorkFeedbackQuestionSet.

[WorkFeedbackQuestionShare](#)

Represents a sharing entry on a WorkFeedbackQuestion.

[WorkFeedbackRequest](#)

Represents a single feedback request on a subject or topic (question) to a single recipient in the feedback application. In the case of offered feedback, WorkFeedbackRequest represents feedback that is offered about a subject. In the performance application, WorkFeedbackRequest represents a request for feedback on a set of questions from a question set, on a subject—for the recipient to complete and submit.

[WorkFeedbackRequestFeed](#)

Represents a single feed item in the feed displayed on the feedback request detail page.

[WorkFeedbackRequestHistory](#)

Represents the history of changes to the values in the fields of a WorkFeedbackRequest.

[WorkFeedbackRequestShare](#)

Represents a sharing entry on a WorkFeedbackRequest.

[WorkFeedbackShare](#)

Represents a sharing entry on a WorkFeedback object.

[WorkGoal](#)

Represents the components of a goal, such as its description and associated metrics. This object has been deprecated as of API version 35.0. Use the [Goal](#) object to query information about Work.com goals.

[WorkGoalCollaborator](#)

Represents collaborators on a WorkGoal object. This doesn't include WorkGoal followers, which is handled by Chatter Feed Follow functionality. This object has been deprecated as of API version 35.0. Use the [Goal](#) object to query information about Work.com goals.

[WorkGoalCollaboratorHistory](#)

Represents the history of changes to the values in the fields in a WorkGoalCollaborator object. Access is read-only.

[WorkGoalFeed](#)

Represents a single feed item in the feed displayed on the goal page for a Goal record. The goal feed shows changes to a goal for fields that are tracked in feeds, posts, and comments about the goal, and updates on metrics. This object has been deprecated as of API version 35.0. Use the [GoalFeed](#) object to query information about feed items for Work.com goals.

[WorkGoalHistory](#)

Represents the history of changes to the values in the fields of a WorkGoal. Access is read-only. This object has been deprecated as of API version 35.0. Use the [GoalHistory](#) object to query historical information for Work.com goals.

[WorkGoalLink](#)

Represents the relationship between two goals (many to many relationship). This object has been deprecated as of API version 35.0. Use the [GoalLink](#) object to query information about the relationship between two Work.com goals.

[WorkGoalShare](#)

Represents a sharing entry on a WorkGoal object. This object has been deprecated as of API version 35.0. Use the [GoalShare](#) object to query information about sharing for Work.com goals.

[WorkOrder](#)

Represents a task or series of tasks to be performed, typically in field service. This object is available in API version 36.0 and later.

Standard Objects

[WorkOrderFeed](#)

Represents a single feed item on a work order record detail page. This object is available in API version 36.0 and later.

[WorkOrderHistory](#)

Represents the history of changes made to tracked fields on a work order. This object is available in API version 36.0 and later.

[WorkOrderLineItem](#)

Represents a subtask on a work order, typically in field service. This object is available in API version 36.0 and later.

[WorkOrderLineItemFeed](#)

Represents a single feed item on a work order line item record detail page. This object is available in API version 36.0 and later.

[WorkOrderLineItemHistory](#)

Represents the history of changes made to tracked fields on a work order line item. This object is available in API version 36.0 and later.

[WorkOrderShare](#)

Represents a sharing entry on a work order. This object is available in API version 36.0 and later.

[WorkPerformanceCycle](#)

Represents feedback that is gathered to assess the performance of a specific set of employees.

[WorkPerformanceCycleFeed](#)

Represents a single feed item in the feed that is displayed on a Work.com Performance detail page.

[WorkPerformanceCycleHistory](#)

Represents the history of changes to the values in the fields of a WorkPerformanceCycle object.

[WorkPerformanceCycleShare](#)

Represents a sharing entry on a WorkPerformanceCycle object.

[WorkReward](#)

Used to store reward codes tied to a Reward Fund. Reward Funds must have at least one WorkReward record.

[WorkRewardFund](#)

Represents a Reward Fund and describes the Reward Fund attributes.

[WorkRewardFundHistory](#)

Represents the history of changes to the values in the fields of a WorkRewardFund object.

[WorkRewardFundShare](#)

Share records for WorkRewardFund.

[WorkRewardFundType](#)

Represents the type of WorkRewardFund object.

[WorkRewardFundTypeHistory](#)

Represents the history of changes to the values in the fields of a WorkRewardFundType object.

[WorkRewardFundTypeShare](#)

Represents a sharing entry on a WorkRewardFundType.

[WorkRewardHistory](#)

Represents the history of changes to the fields of a WorkReward.

[WorkRewardShare](#)

Share records for WorkReward object.

[WorkThanks](#)

Represents the source and message of a thanks post.

[WorkThanksShare](#)

Share records for WorkThanks object.

[WorkType](#)

Represents a type of work to be performed. Work types are templates that can be applied to work order or work order line items. This object is available in API version 38.0 and later.

[WorkTypeFeed](#)

Represents a single feed item on a work type record detail page. This object is available in API version 38.0 and later.

[WorkTypeHistory](#)

Represents the history of changes made to tracked fields on a work type. This object is available in API version 38.0 and later.

[Reusable Terms for API Objects](#)

AcceptedEventRelation

Represents invitees with the status Accepted for a given event.

This object is available in API versions 29.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
EventId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the event.</p>
RelationId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the invitee.</p>
RespondedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	<p>Description Indicates the most recent date and time when the invitee accepted an invitation to the event.</p>
Response	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the content of the response field. Label is Comment.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates whether the invitee is a user, lead or contact, or resource.</p>

Usage

Query invitees who have accepted an invitation to an event

```
SELECT eventId, type, response FROM AcceptedEventRelation WHERE eventid='00UTD000000ZH51A'
```

SEE ALSO:

[DeclinedEventRelation](#)

[UndecidedEventRelation](#)

Account

Represents an individual account, which is an organization or person involved with your business (such as customers, competitors, and partners).

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), merge(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

Customer Portal users can access their own accounts and any account shared with them.

Fields

Field Name	Details
AccountNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Account number assigned to this account (not the unique, system-generated ID assigned during creation). Maximum size is 40 characters.</p>
AccountSource	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The source of the account record. For example, <i>Advertisement</i>, <i>Data.com</i>, or <i>Trade Show</i>. The source is selected from a picklist of available values, which are set by an administrator. Each picklist value can have up to 40 characters.</p>
AnnualRevenue	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Estimated annual revenue of the account.</p>
BillingAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the billing address. Read-only. See Address Compound Fields for details on compound address fields.</p>
BillingCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address of this account. Maximum size is 40 characters.</p>

Field Name	Details
BillingCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address of this account. Maximum size is 80 characters.</p>
BillingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the account's billing address.</p>
BillingGeocodeAccuracy	<p>Type picklist</p> <p>Properties Retrieve, Query, Restricted picklist, Nillable</p> <p>Description Accuracy level of the geocode for the billing address. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
BillingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
BillingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
BillingPostalCode	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address of this account. Maximum size is 20 characters.</p>
BillingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address of this account. Maximum size is 80 characters.</p>
BillingStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the account's billing address.</p>
BillingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street address for the billing address of this account.</p>
CleanStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the record's clean status as compared with Data.com. Values are: Matched, Different, Acknowledged, NotFound, Inactive, Pending, SelectMatch, or Skipped. Several values for CleanStatus display with different labels on the account record detail page. <ul style="list-style-type: none"> • Matched displays as In Sync • Acknowledged displays as Reviewed • Pending displays as Not Compared </p>

Field Name	Details
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Text description of the account. Limited to 32,000 KB.</p>
DunsNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun & Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used by industries and organizations around the world as a global standard for business identification and tracking. Maximum size is 9 characters.</p> <p> Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>
Fax	<p>Type phone</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description Fax number for the account.</p>
Industry	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description An industry associated with this account. Maximum size is 40 characters.</p>
IsCustomerPortal	<p>Type boolean</p>
	<p>Properties Defaulted on create, Filter, Update</p>
	<p>Description Indicates whether the account has at least one contact enabled to use the organization's Customer Portal (<code>true</code>) or not (<code>false</code>). This field is available if Customer Portal is enabled OR Communities is enabled and you have Customer Portal licenses. If you change this field's value from <code>true</code> to <code>false</code>, you can disable up to 100 Customer Portal users associated with the account and permanently delete all of the account's Customer Portal roles and groups. You can't restore deleted Customer Portal roles and groups. This field can be updated in API version 16.0 and later.</p>
	<p>Tip: We recommend that you only update up to 50 contacts simultaneously when changing the accounts on contacts enabled for a Customer Portal or partner portal. We also recommend that you make this update during times outside of your organization's business hours.</p>
IsDeleted	<p>Type boolean</p>
	<p>Properties Defaulted on create, Filter</p>
	<p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPartner	<p>Type boolean</p>
	<p>Properties Defaulted on create, Filter, Update</p>

Field Name	Details
	<p>Description</p> <p>Indicates whether the account has at least one contact enabled to use the organization's partner portal (<code>true</code>) or not (<code>false</code>). This field is available if partner relationship management (partner portal) is enabled OR Communities is enabled and you have partner portal licenses.</p> <p>If you change this field's value from <code>true</code> to <code>false</code>, you can disable up to 15 partner portal users associated with the account and permanently delete all of the account's partner portal roles and groups. You can't restore deleted partner portal roles and groups.</p> <p>Disabling a partner portal user in the Salesforce user interface or the API does not change this field's value from <code>true</code> to <code>false</code>.</p> <p>Even if this field's value is <code>false</code>, you can enable a contact on an account as a partner portal user via the API.</p> <p>This field can be updated in API version 16.0 and later.</p> <p> Tip: We recommend that you only update up to 50 contacts simultaneously when changing the accounts on contacts enabled for a Customer Portal or partner portal. We also recommend that you make this update during times outside of your organization's business hours.</p>
IsPersonAccount	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter</p> <p>Description</p> <p>Read only. Label is Is Person Account. Indicates whether this account has a record type of Person Account (<code>true</code>) or not (<code>false</code>).</p>
Jigsaw	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>References the ID of a company in Data.com. If an account has a value in this field, it means that the account was imported from Data.com. If the field value is <code>null</code>, the account was not imported from Data.com. Maximum size is 20 characters. Available in API version 22.0 and later. Label is Data.com Key.</p> <p> Important: The Jigsaw field is exposed in the API to support troubleshooting for import errors and reimporting of corrected data. Do not modify the value in the Jigsaw field.</p>
LastActivityDate	<p>Type</p> <p>date</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description Value is one of the following, whichever is the most recent:</p> <ul style="list-style-type: none"> • Due date of the most recent event logged against the record. • Due date of the most recently closed task associated with the record.
LastReferencedDate	<p>Type date</p>
	<p>Properties Filter, Nillable, Sort, Update</p>
	<p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p>
	<p>Properties Filter, Nillable, Sort, Update</p>
	<p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
MasterRecordId	<p>Type reference</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description If this object was deleted as the result of a merge, this field contains the ID of the record that was kept. If this object was deleted for any other reason, or has not been deleted, the value is null.</p>
NaicsCode	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The six-digit North American Industry Classification System (NAICS) code is the standard used by business and government to classify business establishments into industries, according to their economic activity for the purpose of collecting, analyzing, and publishing statistical data related to the U.S. business economy. Maximum size is 8 characters.</p>
	<p>Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>

Field Name	Details
NaicsDesc	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A brief description of an organization's line of business, based on its NAICS code. Maximum size is 120 characters.</p> <p> Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Label is Account Name. Name of the account. Maximum size is 255 characters. If the account has a record type of Person Account: <ul style="list-style-type: none"> This value is the concatenation of the <code>FirstName</code>, <code>MiddleName</code>, <code>LastName</code>, and <code>Suffix</code> of the associated person contact. You can't modify this value. </p>
NumberOfEmployees	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Label is Employees. Number of employees working at the company represented by this account. Maximum size is eight digits.</p>
OperatingHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The operating hours associated with the account. Available only if Field Service Lightning is enabled.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>The ID of the user who currently owns this account. Default value is the user logged in to the API to perform the create.</p> <p>If you have set up account teams in your organization, updating this field has different consequences depending on your version of the API:</p> <ul style="list-style-type: none"> For API version 12.0 and later, sharing records are kept, as they are for all objects. For API version before 12.0, sharing records are deleted. For API version 16.0 and later, users must have the “Transfer Record” permission in order to update (transfer) account ownership using this field.
Ownership	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>Ownership type for the account, for example Private, Public, or Subsidiary.</p>
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>ID of the parent object, if any.</p>
Phone	<p>Type</p> <p>phone</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>Phone number for this account. Maximum size is 40 characters.</p>
PhotoUrl	<p>Type</p> <p>url</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Path to be combined with the URL of a Salesforce instance (for example, https://yourInstance.salesforce.com/) to generate a URL to request the social network profile image associated with the account. Generated URL returns an HTTP redirect (code 302) to the social network profile image for the account.</p>

Field Name	Details
	Blank if Social Accounts and Contacts isn't enabled for the organization or if Social Accounts and Contacts is disabled for the requesting user.
Rating	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The account's prospect rating, for example Hot, Warm, or Cold.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
Salutation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Honorific added to the name for use in letters, etc.</p>
ShippingAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the shipping address. Read-only. See Address Compound Fields for details on compound address fields.</p>
ShippingCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details of the shipping address for this account. City maximum size is 40 characters</p>
ShippingCountry	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details of the shipping address for this account. Country maximum size is 80 characters.</p>
ShippingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the account's shipping address.</p>
ShippingGeocodeAccuracy	<p>Type picklist</p> <p>Properties Retrieve, Query, Restricted picklist, Nillable</p> <p>Description Accuracy level of the geocode for the shipping address. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
ShippingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
ShippingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
ShippingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description Details of the shipping address for this account. Postal code maximum size is 20 characters.</p>
ShippingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details of the shipping address for this account. State maximum size is 80 characters.</p>
ShippingStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the account's shipping address.</p>
ShippingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street address of the shipping address for this account. Maximum of 255 characters.</p>
Sic	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Standard Industrial Classification code of the company's main business categorization, for example, 57340 for Electronics. Maximum of 20 characters.</p>
SicDesc	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A brief description of an organization's line of business, based on its SIC code. Maximum length is 80 characters.</p>
Site	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name of the account's location, for example <code>Headquarters</code> or <code>London</code>. Label is Account Site. Maximum of 80 characters.</p>
TickerSymbol	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The stock market symbol for this account. Maximum of 20 characters.</p>
Tradestyle	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A name, different from its legal name, that an organization may use for conducting business. Similar to "Doing business as" or "DBA". Maximum length is 255 characters.</p> <p> Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Type of account, for example, Customer, Competitor, or Partner.</p>
Website	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The website of this account. Maximum of 255 characters.</p>
YearStarted	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The date when an organization was legally established. Maximum length is 4 characters.</p> <p> Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>

IsPersonAccount Fields

These fields are the subset of person account fields that are contained in the child person contact record of each person account. If the `IsPersonAccount` field has the value `false`, the following fields have a null value and can't be modified. If `true`, the fields contain the value indicated in the Description column in the following table and can be modified.

Person accounts are not enabled by default.

Field Name	Details
FirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description First name of the person for a person account. Maximum size is 40 characters.</p>
LastName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Last name of the person for a person account. Required if the record type is a person account record type. Maximum size is 80 characters.</p>
MiddleName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Middle name of the person for a person account. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
PersonAssistantName	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p>

Field Name	Details
	<p>Description The person account's assistant name. Label is Assistant. Maximum size is 40 characters.</p>
PersonAssistantPhone	<p>Type phone</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The person account's assistant phone. Label is Asst. Phone. Maximum size is 40 characters.</p>
PersonBirthDate	<p>Type date</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The assistant name. Label is Birthdate.</p> <p> Note: The year portion of the PersonBirthDate field is ignored in filter criteria, including report filters, list view filters, and SOQL queries. For example, the following SOQL query returns person accounts with birthdays later in the year than today:</p> <pre>SELECT FirstName, LastName, PersonBirthDate FROM Account WHERE Birthdate > TODAY</pre>
PersonContactId	<p>Type reference</p> <p>Properties Filter, Nillable, Update</p> <p>Description The ID for the contact associated with this person account. Label is Contact ID.</p>
PersonDepartment	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The department. Label is Department. Maximum size is 80 characters.</p>
PersonEmail	<p>Type email</p> <p>Properties Create, Filter, Nillable, Update</p>

Field Name	Details
	Description
	Email address for this person account. Label is Email .
PersonEmailBouncedDate	Type
	dateTime
	Properties
	Create, Filter, Nullable, Update
	Description
	If bounce management is activated and an email sent to the person account bounces, the date and time the bounce occurred.
PersonEmailBouncedReason	Type
	string
	Properties
	Create, Filter, Nullable, Update
	Description
	If bounce management is activated and an email sent to the person account bounces, the reason the bounce occurred
PersonHasOptedOutOfEmail	Type
	boolean
	Properties
	Create, Filter, Nullable, Update
	Description
	Indicates whether the person account has opted out of email (<code>true</code>) or not (<code>false</code>). Label is Email Opt Out .
PersonHomePhone	Type
	phone
	Properties
	Create, Filter, Nullable, Update
	Description
	The home phone number for this person account. Label is Home Phone .
PersonLastCURequestDate	Type
	dateTime
	Properties
	Create, Filter, Nullable, Update
	Description
	The last date that this person account was requested. Label is Last Stay-in-Touch Request Date .

Field Name	Details
PersonLastCUUpdateDate	Type dateTime
	Properties Create, Filter, Nillable, Update
	Description The last date a person account was updated. Label is Last Stay-in-Touch Save Date .
PersonLeadSource	Type picklist
	Properties Create, Filter, Nillable, Update
	Description The person account's lead source. Label is Lead Source .
PersonMailingAddress	Type address
	Properties Filter, Nillable
	Description The compound form of the person account mailing address. Read-only. See Address Compound Fields for details on compound address fields.
PersonMailingCity	Type string
	Properties Create, Filter, Nillable, Update
	Description Details about the person account's mailing city. Labels are Mailing City , Mailing Country , Postal Code , and State . Maximum size for city and country is 40 characters. Maximum size for postal code and state is 20 characters.
PersonMailingGeocodeAccuracy	Type picklist
	Properties Retrieve, Query, Restricted picklist, Nillable
	Description Accuracy level of the geocode for the person's mailing address. See Compound Field Considerations and Limitations for details on geolocation compound fields.
PersonMailingLatitude	Type double

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Used with <code>PersonMailingLongitude</code> to specify the precise geolocation of a person account's mailing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
PersonMailingLongitude	<p>Type double</p>
	<p>Properties Create, Filter, , Nillable, Sort, Update</p>
	<p>Description Used with <code>PersonMailingLatitude</code> to specify the precise geolocation of a person account's mailing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations on page 42 for details on geolocation compound fields.</p>
PersonMailingStreet	<p>Type textarea</p>
	<p>Properties Create, Filter, Nillable, Update</p>
	<p>Description The mailing street address for this person account. Label is Mailing Street. Maximum size is 255 characters.</p>
PersonMobilePhone	<p>Type phone</p>
	<p>Properties Create, Filter, Nillable, Update</p>
	<p>Description The mobile phone number for this person account. Label is Mobile.</p>
<ul style="list-style-type: none"> • <code>PersonOtherCity</code> • <code>PersonOtherCountry</code> • <code>PersonOtherPostalCode</code> • <code>PersonOtherState</code> 	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p>
	<p>Description Details about the alternate address for this person account. Labels are Other City, Other Country, Other Zip/Postal Code, and Other State.</p>

Field Name	Details
<ul style="list-style-type: none"> • PersonOtherCountryCode • PersonOtherStateCode 	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country or state code for the alternate address of the person account.</p>
PersonOtherLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>PersonOtherLongitude</code> to specify the precise geolocation of a person account's alternate address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
PersonOtherLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>PersonOtherLatitude</code> to specify the precise geolocation of a person account's alternate address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
PersonOtherPhone	<p>Type phone</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The alternate phone number for this person account. Label is Other Phone.</p>
PersonOtherStreet	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The person account's alternate street address. Label is Other Street.</p>

Field Name	Details
PersonTitle	<p>Type string</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The person account's title. Label is Title. Maximum size is 80 characters.</p>
Suffix	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Name suffix of the person for a person account. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>



Note: If you are importing Account data into Salesforce and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself.

Usage

Use this object to query and manage accounts in your organization. Client applications can create, update, delete, or query Attachment records associated with an account via the API.

Client applications can also create or update account objects by converting a Lead via the `convertLead()` call.

If the values in the `IsPersonAccount` Fields are not null, you can't change `IsPersonAccount` to `false`, or an error occurs.

SEE ALSO:

- [AccountShare](#)
- [AccountTeamMember](#)
- [Person Account Record Types](#)
- [AccountHistory](#)

AccountContactRelation

Represents a relationship between a contact and one or more accounts.

This object is available in API version 37.0.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the account that is related to the contact.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the contact that is related to the account.</p>
EndDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The date a relationship between a contact and account began. Use with the Start Date to keep a history of the relationship.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether relationship is active (<code>true</code>) or not (<code>false</code>).</p>
IsDirect	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the account associated with the contact is the contact's primary account (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
Roles	<p>Type multipicklist</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The contact's participating role in the account. Values are <code>Business User</code>, <code>Decision Maker</code>, <code>Economic Buyer</code>, <code>Economic Decision Maker</code>, <code>Evaluator</code>, <code>Executive Sponsor</code>, <code>Influencer</code>, <code>Technical Buyer</code>, and <code>Other</code>.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The date a relationship between a contact and account ended. Use with the <code>EndDate</code> to keep a history of the relationship.</p>

Usage

Use this object to associate a single contact record to multiple account records so you can easily track the relationships between the people and businesses they work with.

AccountCleanInfo

Stores the metadata Data.com Clean uses to determine an account record's clean status. AccountCleanInfo helps you automate the cleaning or related processing of account records.

Account Clean Info provides a snapshot of the data in your Salesforce account record and its matched Data.com record at the time the Salesforce record was cleaned.

Account Clean Info includes a number of bit vector fields, whose component fields each correspond to individual object fields and provide related data or status information about those fields. For example, the bit vector field `IsDifferent` has an `IsDifferentState` field. If the `IsDifferentState` field's value is `False`, that means the `State` field value is *the same* on the Salesforce account record and its matched Data.com record.

AccountCleanInfo bit vector fields include:

- `CleanedBy` indicates who (a user) or what (a Clean job) cleaned the account record.
- `IsDifferent` indicates whether or not a field on the account record has a value that differs from the corresponding field on the matched Data.com record.
- `IsFlaggedWrong` indicates whether or not a field on the account record has a value that is flagged as wrong to Data.com.
- `IsReviewed` indicates whether or not a field on the account record is in a `Reviewed` state, which means that the value was reviewed but not accepted.

Their individual bits are defined here.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique, system-generated ID assigned when the account record was created.</p>
AccountSite	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Information about the account's location, such as single location, headquarters, or branch.</p>
Address	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
AnnualRevenue	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Estimated annual revenue of the account.</p>
City	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address of the account.</p>

Field Name	Details
CleanedByJob	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account record was cleaned by a Data.com Clean job (<code>true</code>) or not (<code>false</code>).</p>
CleanedByUser	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account record was cleaned by a Salesforce user (<code>true</code>) or not (<code>false</code>).</p>
CompanyName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the company.</p>
CompanyStatusDataDotCom	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The status of the company per Data.com. Values are: Company is In Business per Data.com or Company is Out of Business per Data.com.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address of the account.</p>
DandBCompanyDunsNumber	<p>Type string</p>

Field Name	Details
	Properties Filter, Group, Nillable, Sort
	Description The D-U-N-S Number on the D&B Company record (if any) that is linked to the account.
DataDotComId	Type string
	Properties Filter, Group, Nillable, Sort
	Description The ID Data.com maintains for the company.
Description	Type textarea
	Properties Nillable
	Description A description of the account.
DunsNumber	Type string
	Properties Filter, Group, Nillable, Sort
	Description The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun & Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used by industries and organizations around the world as a global standard for business identification and tracking.
DunsRightMatchConfidence	Type int
	Properties Filter, Group, Nillable, Sort
	Description The account's DUNSRight confidence code.
DunsRightMatchGrade	Type string
	Properties Filter, Group, Nillable, Sort

Field Name	Details
	Description
	The account's DUNSRight match grade.
Fax	Type
	phone
	Properties
	Filter, Group, Nillable, Sort
	Description
	The account's fax number.
Industry	Type
	picklist
	Properties
	Filter, Group, Nillable, Sort
	Description
	The industry the account belongs to.
IsDifferentAccountSite	Type
	boolean
	Properties
	Filter
	Description
	Indicates whether the account's <code>AccountSite</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentAnnualRevenue	Type
	boolean
	Properties
	Filter
	Description
	Indicates whether the account's <code>AnnualRevenue</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentCity	Type
	boolean
	Properties
	Filter
	Description
	Indicates whether the account's <code>City</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).

Field Name	Details
IsDifferentCompanyName	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>AccountName</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountry	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Country</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountryCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Country Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentDandBCompanyDunsNumber	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>DandBCompanyID</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentDescription	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Description</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentDunsNumber	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>DunsNumber</code> field value is different from the D-U-N-S Number on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentFax	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Fax</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentIndustry	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Industry</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentNaicsCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>NaicsCode</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentNaicsDescription	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>NaicsDescription</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentNumberOfEmployees	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>NumberOf Employees</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentOwnership	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Ownership</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentPhone	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Phone</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentPostalCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>PostalCode</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentSic	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>SIC</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentSicDescription	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>SicDescription</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentState	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>State</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentStateCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>State Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentStreet	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>State</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentTickerSymbol	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>TickerSymbol</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentTradestyle	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Tradestyle</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentWebsite	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Website</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentYearStarted	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>YearStarted</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsFlaggedWrongAccountSite	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>AccountSite</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
IsFlaggedWrongAddress	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>Address</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
IsFlaggedWrongAnnualRevenue	<p>Type boolean</p>

Field Name	Details
	Properties
	Filter, Update
	Description
	Indicates whether the account's AnnualRevenue field value is flagged as wrong to Data.com (true) or not (false).
IsFlaggedWrongCompanyName	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's CompanyName field value is flagged as wrong to Data.com (true) or not (false).
IsFlaggedWrongDescription	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's Description field value is flagged as wrong to Data.com (true) or not (false).
IsFlaggedWrongDunsNumber	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's DunsNumber field value is flagged as wrong to Data.com (true) or not (false).
IsFlaggedWrongFax	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's Fax field value is flagged as wrong to Data.com (true) or not (false).
IsFlaggedWrongIndustry	Type
	boolean
	Properties
	Filter, Update

Field Name	Details
	<p>Description Indicates whether the account's <code>Industry</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongNaicsCode</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NaicsCode</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongNaicsDescription</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NaicsDescription</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongNumberOfEmployees</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NumberOfEmployees</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongOwnership</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>Ownership</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongPhone</code>	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	Description
	Indicates whether the account's <code>Phone</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongSic	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>Sic</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongSicDescription	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>SicDescription</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongTickerSymbol	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>TickerSymbol</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongTradestyle	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>Tradestyle</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongWebsite	Type
	boolean
	Properties
	Filter, Update

Field Name	Details
	Description
	<p>Indicates whether the account's <code>Website</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
IsFlaggedWrongYearStarted	Type
	boolean
	Properties
	Filter, Update
	Description
	<p>Indicates whether the account's <code>YearStarted</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
IsInactive	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	<p>Indicates whether the account has been reported to Data.com as <code>Inactive</code> (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedAccountSite	Type
	boolean
	Properties
	Filter, Update
	Description
	<p>Indicates whether the account's <code>AccountSite</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedAddress	Type
	boolean
	Properties
	Filter, Update
	Description
	<p>Indicates whether the account's <code>Address</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedAnnualRevenue	Type
	boolean
	Properties
	Filter, Update

Field Name	Details
	Description
	Indicates whether the account's AnnualRevenue field value is in a Reviewed state (true) or not (false).
IsReviewedCompanyName	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's CompanyName field value is in a Reviewed state (true) or not (false).
IsReviewedDandBCompanyDunsNumber	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's DandBCompanyID field value is in a Reviewed state (true) or not (false).
IsReviewedDescription	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's Description field value is in a Reviewed state (true) or not (false).
IsReviewedDunsNumber	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's DunsNumber field value is in a Reviewed state (true) or not (false).
IsReviewedFax	Type
	boolean
	Properties
	Filter, Update

Field Name	Details
	<p>Description Indicates whether the account's <code>Fax</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedIndustry</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>Industry</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedNaicsCode</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NaicsCode</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedNaicsDescription</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NaicsDescription</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedNumberOfEmployees</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>NumberOfEmployees</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedOwnership</code>	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	<p>Description Indicates whether the account's <code>Ownership</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedPhone	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>Phone</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedSic	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>Sic</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedSicDescription	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>SicDescription</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedTickerSymbol	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the account's <code>TickerSymbol</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
IsReviewedTradestyle	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	Description
	Indicates whether the account's <code>Tradestyle</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).
IsReviewedWebsite	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>Website</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).
IsReviewedYearStarted	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the account's <code>YearStarted</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).
LastMatchedDate	Type
	dateTime
	Properties
	Filter, Sort
	Description
	The date the account record was last matched and linked to a Data.com record.
LastStatusChangedById	Type
	reference
	Properties
	Filter, Group, Nullable, Sort
	Description
	The ID of who or what last changed the record's <code>Clean Status</code> field value: a Salesforce user or a Clean job.
LastStatusChangedDate	Type
	dateTime
	Properties
	Filter, Nullable, Sort
	Description
	The date on which the record's <code>Clean Status</code> field value was last changed.

Field Name	Details
Latitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with <code>Longitude</code> to specify the precise geolocation of a billing address. Data not currently provided.</p>
Longitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with <code>Latitude</code> to specify the precise geolocation of a billing address. Data not currently provided.</p>
NaicsCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The six-digit North American Industry Classification System (NAICS) code is the standard used by business and government to classify business establishments into industries, according to their economic activity for the purpose of collecting, analyzing, and publishing statistical data related to the U.S. business economy.</p>
NaicsDescription	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A brief description of an organization's line of business, based on its NAICS code.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description Field label is Account Clean Info Name. The name of the account. Maximum size is 255 characters.</p>

Field Name	Details
NumberOfEmployees	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of employees working at the account.</p>
Ownership	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Ownership type for the account, for example Private, Public, or Subsidiary.</p>
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The phone number for the account.</p>
PostalCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address of the account.</p>
Sic	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Standard Industrial Classification code of the company's main business categorization, for example, 57340 for Electronics.</p>
SicDescription	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	A brief description of an organization's line of business, based on its SIC code.
State	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the account.
Street	Type
	textarea
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the account.
TickerSymbol	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The stock market symbol for the account.
Tradestyle	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	A name, different from its legal name, that an organization can use for conducting business. Similar to "Doing business as" (DBA).
Website	Type
	url
	Properties
	Filter, Group, Nillable, Sort
	Description
	The website of the account.
YearStarted	Type
	string

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The year the company was established or the year when current ownership or management assumed control of the company.</p>

Usage

Administrators can modify a limited set of AccountCleanInfo fields from the Account Clean Info page.

Developers can create triggers that read the Account Clean Info fields to help automate the cleaning or related processing of account records. For example, you might create a trigger that reads the `Clean Status` field on the Account object. If an account record's `Clean Status` field value is `Different` but the record has no `Billing Street` value, the trigger could update the record's status to `Not Compared`.

Create triggers that read AccountCleanInfo fields to help automate the cleaning or related processing of account records. For example:

- Keep account records' status `InSync` if the only difference from matched records is the `Phone` format (for example, `(415) 353-8000` on the account record versus `415 353 8000` on the matched Data.com record).

```
trigger AccountPhoneTrigger on Account (before update) {
    for (Account account: Trigger.new) {
        Account oldAccount = Trigger.oldMap.get(account.ID);
        if (account.CleanStatus == 'Different') {
            List <AccountCleanInfo> cleanInfo = [Select Id, IsDifferentPhone,
IsReviewedPhone, Phone from AccountCleanInfo where AccountId = :account.Id];
            if (cleanInfo.size() > 0 && cleanInfo[0].IsDifferentPhone &&
cleanInfo[0].Phone.StartsWith('+')) {
                // if Data.com phone number is marked Different but starts with '+',
                ignore this
                // and set the status to "Reviewed"
                AccountCleanInfo cleanInfoToUpdate = new AccountCleanInfo();
                cleanInfoToUpdate.Id = cleanInfo[0].Id;
                cleanInfoToUpdate.IsReviewedPhone = true;
                update cleanInfoToUpdate;
                account.CleanStatus = 'Reviewed';
            }
        }
    }
}
```

- Create a customized set of `Industry` field values for accounts. Use triggers to map values from fields on imported or cleaned records onto a standard set of values.
- Read the `CleanStatus` field value on the Account object. If that value is `Different`, but a Salesforce record has no street address value, update the record's status to `Not Compared`.

AccountContactRole

Represents the role that a Contact plays on an Account.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Account.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. ID of the Contact associated with this account.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Specifies whether the Contact plays the primary role on the Account (<code>true</code>) or not (<code>false</code>). Note that each account has only one primary contact role. Label is Primary. Default value is <code>false</code>.</p>
Role	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name of the role played by the Contact on this Account, such as Decision Maker, Approver, Buyer, and so on. Must be unique—there can't be multiple records in which the <code>AccountId</code>, <code>ContactId</code>, and <code>Role</code> values are identical. Different contacts can play the same role on the same account. A contact can play different roles on the same account.</p>

Usage

Use this object to define the role that a Contact plays on a given Account within the context of a specific Opportunity.

SEE ALSO:

[Account](#)

[Contact](#)

AccountFeed

Represents a single feed item on an account record detail page. This object is available in API version 18.0 and later.

An account feed shows changes to an account record for fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to accounts in Salesforce.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Account object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of AccountFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p>

Field	Details
	Description
	<p>Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	Type
	textarea
	Properties
	Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	Type
	int
	Properties
	Group, Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	Type
	dateTime
	Properties
	Defaulted on create, Filter, Sort

Field	Details
	<p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
isRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags: <ul style="list-style-type: none"> • <p> </p>

Field	Details
	<p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>

Field	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL. PollPost—a poll posted on a feed. ProfileSkillPost—generated when a skill is added to a user's Chatter profile. QuestionPost—generated when a user posts a question. ReplyPost—generated when Chatter Answers posts a reply. RipplePost—generated when a user creates a Thanks badge in Work.com.

Field	Details
	<ul style="list-style-type: none">• <code>TextPost</code>—a direct text entry on a feed.• <code>TrackedChange</code>—a change or group of changes to a tracked field.• <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none">• <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for an account record.

SEE ALSO:

[Account](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

AccountHistory

Represents the history of changes to the values in the fields of an account. This object is available in versions 11.0 and later.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Account. Label is Account ID.</p>

Field	Details
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to an account.

This object respects field level security on the parent object.

SEE ALSO:

[Account](#)

AccountOwnerSharingRule

Represents the rules for sharing an account with a User other than the owner.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All (This value isn't valid for creating or updating.)
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group for all child cases. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit
ContactAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description A value that represents the type of access granted to the target Group, UserRole, or User for any associated contacts. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p> Note: When DefaultContactAccess is set to Controlled by Parent, you can't create or update this field.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. An Account owned by a User in the source Group triggers the rule to give access.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group for any associated Opportunity. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit
Name	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the User or Group being granted access.</p>

Usage

Use this object to manage the sharing rules for accounts. General sharing and territory management-related sharing use this object. For example, the following code creates an account owner sharing rule between two public groups, which can also contain portal users.

```
AccountOwnerSharingRule rule = new AccountOwnerSharingRule();
rule.setName("RuleName"); // Set the sharing rule name
rule.setDeveloperName("RuleDeveloperName"); // Set the sharing rule developer name
rule.setGroupId("00Gx000000000000"); // Set the group of users to share records from
rule.setUserOrGroupId("00Gx000000000001"); // Set the group of users to share records to
rule.setAccountAccessLevel("Edit");
rule.setOpportunityAccessLevel("Read");
rule.setCaseAccessLevel("None");
connection.create(rule);
```

SEE ALSO:

[Account](#)

[AccountShare](#)

[Metadata API Developer Guide: SharingRules](#)

AccountPartner

This read-only object represents a partner relationship between two Account objects. It is automatically created when a Partner object is created for a partner relationship between two accounts.

 **Note:** This object is completely distinct and independent of Account records that have been enabled for the partner portal.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AccountFromId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the main Account in the partner relationship.</p>
AccountToId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort,</p> <p>Description ID of the partner Account in the partner relationship.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the AccountPartner is the main account's primary partner (<code>true</code>) or not (<code>false</code>).</p>
OpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the opportunity in a partner relationship with the related account.</p>

Field	Details
Role	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The UserRole that the partner Account has on the main Account. For example, Consultant or Distributor.</p>

Creating an Account-Account Partner Relationship

When you create a partner relationship between two accounts (when you create a Partner object and specify the `AccountFromId`), the API automatically creates two AccountPartner objects, one for the forward relationship and one for the reverse. For example, if you create a Partner object with "Acme, Inc." as the `AccountFromId` and "Acme Consulting" as the `AccountToId`, the API automatically creates two AccountPartner objects:

- The forward relationship AccountPartner with "Acme, Inc." as the `AccountFromId` and "Acme Consulting" as the `AccountToId`.
- The reverse relationship AccountPartner with "Acme Consulting" as the `AccountFromId` and "Acme, Inc." as the `AccountToId`.
- The value of the `Role` field in the reverse relationship AccountPartner is set to the PartnerRole object `ReverseRole` value associated with the value of the `Role` field in the forward relationship AccountPartner.

This mapping allows the API to manage the objects and their relationship efficiently.

SEE ALSO:

[Partner](#)
[OpportunityPartner](#)

AccountShare

Represents a sharing entry on an Account.

Supported Calls

`describeSObjects()`, `create()`, `delete()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to the Account. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All (This value isn't valid for create or update calls.) <p>This field must be set to an access level that is at least equal to the organization's default Account access level. In addition, either this field, the OpportunityAccessLevel field, or the CaseAccessLevel field must be set higher than the organization's default access level.</p>
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Account associated with this sharing entry. This field can't be updated.</p>
CaseAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to cases associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default CaseAccessLevel. This field can't be updated via the API if the AccountAccessLevel field is set to All. You can't update this field for the associated account owner via the API. You must update the account owner's CaseAccessLevel via the Salesforce user interface.</p>
ContactAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>Level of access that the User or Group has to contacts associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default <code>ContactAccessLevel</code>. This field can't be updated via the API if the <code>ContactAccessLevel</code> field is set to "Controlled by Parent." You can't update this field for the associated account owner using the API. You must update the account owner's <code>ContactAccessLevel</code> via the Salesforce user interface.</p>
<code>IsDeleted</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
<code>OpportunityAccessLevel</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to opportunities associated with the Account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default opportunity access level. This field can't be updated via the API if the <code>AccountAccessLevel</code> field is set to <code>All</code>. You can't use the API to update this field for the associated Account owner. You must update the Account owner's <code>opportunityAccessLevel</code> via the Salesforce user interface.</p>
<code>RowCause</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field	Details
Description	
Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.	
Valid values include:	
<ul style="list-style-type: none"> • <code>Manual</code>—The User or Group has access because a User with “All” access manually shared the Account with the user or group. • <code>ImplicitParent</code>—The User or Group has access because they’re the owner of or have sharing access to records related to the account, such as opportunities, cases, contacts, contracts, or orders. • <code>Owner</code>—The User is the owner of the Account • <code>Team</code>—The User or Group has team access (is an <code>AccountTeamMember</code>). • <code>Rule</code>—The User or Group has access via an Account sharing rule. 	
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the Account. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit Account records owned by other users.

If you attempt to create an `AccountShare` record that matches an existing record, the request updates any modified fields and returns the existing record.

For example, the following code finds all accounts owned by a user and manually shares them to a portal user.

```
QueryResult result = conn.query("SELECT Id FROM Account WHERE OwnerId = '005D0000001LPFB'");
// Create a new AccountShare object
List<AccountShare> shares = new ArrayList<AccountShare>();
for (SObject rec : result.getRecords()) {
    AccountShare share = new AccountShare();
    share.setAccountId(rec.getId());
    //Set the portal user Id to share the accounts with
    share.setUserOrGroupId("003D000000QA8T1");
    share.setAccountAccessLevel("Edit");
    share.setOpportunityAccessLevel("Read");
    share.setCaseAccessLevel("Edit");
    shares.add(share);
}
conn.create(shares.toArray(new AccountShare[shares.size()]));
```

This code shares the accounts that the user owns at the time, but not those accounts that are owned later. For these types of shares, use an owner-based sharing rule, such as `AccountOwnerSharingRule`.

If an account is shared in multiple ways with a user, you don't always see multiple sharing records. If a user has access to an account for one or more of the following RowCause values, the records in the AccountShare object are compressed into one record with the highest level of access.

- ImplicitParent
- Manual
- Owner

SEE ALSO:

[Account](#)
[CaseShare](#)
[LeadShare](#)
[OpportunityShare](#)

AccountTag

Associates a word or short phrase with an Account.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
ItemId	Type reference Properties Create, Filter Description ID of the tagged item.
Name	Type string Properties Create, Filter Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.
TagDefinitionId	Type reference

Field Name	Details
	<p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching <code>OwnerId</code>.

Usage

AccountTag stores the relationship between its parent TagDefinition and the Account being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

AccountTeamMember

Represents a User who is a member of an Account team.

See also `UserAccountTeamMember`, which represents a User who is on the default account team of another user.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), update(), upsert()
```

Special Access Rules

- This object is available only for Enterprise, Unlimited, and Performance Edition users who have enabled the account team functionality.
- Customer Portal users can't access this object.

Fields

Field Name	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Level of access that the User has to the Account. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All <p>This field must be set to an access level that is at least equal to the organization's default Account access level. In addition, the user's AccountAccessLevel, ContactAccessLevel, OpportunityAccessLevel, or CaseAccessLevel field must be set higher than the organization's default access level.</p>
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Account to which this user is a team member. Must be a valid account ID.</p>
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Level of access that the User has to cases associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default case access level. In addition, the user's AccountAccessLevel, ContactAccessLevel, OpportunityAccessLevel, or CaseAccessLevel field must be set higher than the organization's default access level. This field is available in API version 37.0 and later.</p>

Field Name	Details
ContactAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Level of access that the User has to contacts associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default contact access level. In addition, the user's AccountAccessLevel, ContactAccessLevel, OpportunityAccessLevel, or CaseAccessLevel field must be set higher than the organization's default access level. This field is available in API version 37.0 and later.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p> <p> Note: An AccountTeamMember record that is deleted is not moved to the Recycle Bin. A deleted AccountTeamMember record can't be undeleted unless the record was cascade-deleted when deleting a related Account. For directly deleted AccountTeamMember records, don't use the isDeleted field to detect deleted records in SOQL queries or <code>queryAll()</code> calls. Instead, use <code>getDeleted()</code>.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Level of access that the User has to opportunities associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is at least equal to the organization's default opportunity access level. In addition, the user's AccountAccessLevel,</p>

Field Name	Details
	ContactAccessLevel, OpportunityAccessLevel, or CaseAccessLevel field must be set higher than the organization's default access level. This field is available in API version 37.0 and later.
PhotoURL	<p>Type URL</p> <p>Properties Filter, Nillable, Sort, Group</p> <p>Description Read only. Retrieves the users Chatter photo URL. This field is available in API version 37.0 and later.</p>
TeamMemberRole	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Role associated with this team member. One of the valid team member roles defined for your organization. Label is Team Role.</p>
Title	<p>Type string</p> <p>Properties Filter, Nillable, Sort, Group</p> <p>Description Read only. Retrieves the user's title. This field is available in API version 37.0 and later.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the User who is a member of this account team. Must be a valid User ID.</p>

Usage

Use this object to manage the team members of a particular Account and to specify team member roles for those users on that account.

SEE ALSO:

[Account](#)

AccountTerritoryAssignmentRule

An account assignment rule that assigns accounts to territories based on account fields. Only available if territory management has been enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
BooleanFilter	<p>Type string</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Advanced filter conditions that were specified for the rule in the online application. For example, "(1 AND 2) OR 3."</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the rule is active (<code>true</code>) or inactive (<code>false</code>). Via the API, active rules run automatically when new accounts are created and existing accounts are edited. The exception is when the <code>IsExcludedFromRealign</code> field on an account is <code>true</code>, which prevents account assignment rules from evaluating that account.</p>
IsInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the rule is an inherited rule (<code>true</code>) or a local rule (<code>false</code>). An inherited rule also acts upon territories below it in the territory hierarchy. A local rule is created at the immediate territory and only impacts the immediate territory.</p>

Field	Details
Name	Type string Properties Create, Filter, Update Description A name for the rule. Limit is 80 characters.
TerritoryId	Type reference Properties Create, Filter, Update Description ID of the territory where accounts that satisfy this rule are assigned.

Usage

A territory will not have any accounts (with the exception of manually assigned accounts) unless at least one account assignment rule is active for the territory.

SEE ALSO:

[AccountTerritoryAssignmentRuleItem](#)

[Territory](#)

[UserTerritory](#)

AccountTerritoryAssignmentRuleItem

A row of selection criteria for an AccountTerritoryAssignmentRule object. Only available if territory management has been enabled for your organization.

AccountTerritoryAssignmentRuleItem can be created or deleted if the BooleanFilter field on its corresponding AccountTerritoryAssignmentRule object is a null value.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The standard or custom account field to use as a criteria.</p>
Operation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The criteria to apply, such as "equals" or "starts with."</p>
RuleID	<p>Type reference</p> <p>Properties Create, Filter, Update</p> <p>Description ID of the associated AccountTerritoryAssignmentRule.</p>
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Update</p> <p>Description The order in which this row is evaluated compared to other AccountTerritoryAssignmentRuleItem objects for the given AccountTerritoryAssignmentRule.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The field value(s) to evaluate, such as 94105 if the Field is Billing Zip/Postal Code.</p>

Usage

- Both standard and custom account fields can be used as criteria for account assignment rules.
- A territory will not have any accounts (with the exception of manually assigned accounts) unless at least one account assignment rule is active for the territory.

SEE ALSO:

[AccountTerritoryAssignmentRule](#)

[Territory](#)

[UserTerritory](#)

AccountTerritorySharingRule

Represents the rules for sharing an Account within a Territory.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none">ReadEditAll
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field	Details
	<p>Description</p> <p>A value that represents the type of access granted to the target group for all child cases of the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit
ContactAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description</p> <p>A value that represents the type of access granted to the target group for all related contacts on the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p> Note: This field is read only.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description</p> <p>The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Accounts owned by users in the source territory trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to <code>Label</code> on the user interface.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target group for all opportunities associated with the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the user or group being given access, or, if a territory ID, the users assigned to that territory.</p>

Usage

Use this object to manage the sharing rules for a particular object. General sharing and Territory-related sharing use this object.

SEE ALSO:

[Account](#)

[AccountShare](#)

ActionLinkGroupTemplate

Action link templates let you reuse action link definitions and package and distribute action links. An action link is a button on a feed element. Clicking on an action link can take a user to another Web page, initiate a file download, or invoke an API call to an external server or Salesforce. Use action links to integrate Salesforce and third-party services into the feed. Every action link belongs to an action link group and action links within the group are mutually exclusive. This object is available in API version 33.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Only users with the “Customize Application” permission can modify or delete this object.

Fields

Field Name	Details
Category	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The location of the action link group within the feed element. Values are:</p> <ul style="list-style-type: none">• Primary—The action link group is displayed in the body of the feed element.• Overflow—The action link group is displayed in the overflow menu of the feed element.
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the action link group template to use in code.</p>

Field Name	Details
ExecutionsAllowed	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The number of times an action link can be executed. Values are:</p> <ul style="list-style-type: none"> Once—An action link can be executed only once across all users. OncePerUser—An action link can be executed only once for each user. Unlimited—An action link can be executed an unlimited number of times by each user. If the action link's <code>actionType</code> is <code>Api</code> or <code>ApiAsync</code>, you can't use this value.
HoursUntilExpiration	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The number of hours from when the action link group is created until it's removed from associated feed elements and can no longer be executed. The maximum value is 8,760.</p>
IsPublished	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, the action link group template is published. Action link group templates shouldn't be published until at least one <code>ActionLinkTemplate</code> is associated with it. Once set to <code>true</code>, this can't be set back to <code>false</code>.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the <code>MasterLabel</code>.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description The name of the action link group template.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

Define action link templates in Setup and use `ConnectApi` in Apex or Chatter REST API to instantiate action links from the templates and to post feed elements with the action links.

If you delete a published action link group template, you delete all related action link information which includes deleting all action links that were instantiated using the template from feed items.

ActionLinkTemplate

Action link templates let you reuse action link definitions and package and distribute action links. An action link is a button on a feed element. Clicking an action link can take a user to another Web page, initiate a file download, or invoke an API call to an external server or Salesforce. Use action links to integrate Salesforce and third-party services into the feed. This object is available in API version 33.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Only users with the “Customize Application” permission can modify or delete this object.

Fields

Field Name	Details
ActionLinkGroupTemplateId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the ActionLinkGroupTemplate with which this action link template is associated.</p>
ActionUrl	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The action link URL. For example, a <code>Ui</code> action link URL is a Web page. A <code>Download</code> action link URL is a link to the file to download. <code>Ui</code> and <code>Download</code> action link URLs are provided to clients. An <code>Api</code> or <code>ApiAsync</code> action link URL is a REST resource. <code>Api</code> and <code>ApiAsync</code> action link URLs aren't provided to clients. Links to Salesforce can be relative. All other links must be absolute and start with <code>https://</code>. Links to resources hosted on Salesforce servers can be relative, starting with a <code>/</code>. All other links must be absolute and start with <code>https://</code>. This field can contain context variables and binding variables in the form <code>{!Bindings.key}</code>, for example, <code>https://www.example.com/{!Bindings.itemId}</code>. Set the binding variable's value when you instantiate the action link group from the template.</p>
Headers	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Template for the HTTP headers sent when corresponding action links are invoked. This field can be used only for <code>Api</code> and <code>ApiAsync</code> action links. This field can contain context variables and binding variables in the form <code>{!Bindings.key}</code>.</p>

Field Name	Details
<code>IsConfirmationRequired</code>	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, a confirmation dialog appears before the action is executed.</p>
<code>IsGroupDefault</code>	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, action links derived from this template are the default or primary action in their action groups. There can be only one default action per action group.</p>
<code>Label</code>	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A custom label to display on the action link button. If none of the <code>LabelKey</code> values make sense for an action link, use a custom label. Set the <code>LabelKey</code> field to <code>None</code> and enter a label name in the <code>Label</code> field. Action links have four states: new, pending, success, and failed. These strings are appended to the label for each state:</p> <ul style="list-style-type: none"> • <code>Label</code> • <code>Label Pending</code> • <code>Label Success</code> • <code>Label Failed</code> <p>For example, if the value of <code>Label</code> is "Call Home," the values of the four action link states are: Call Home, Call Home Pending, Call Home Success, and Call Home Failed. If <code>LabelKey</code> has any value other than <code>None</code>, the <code>Label</code> field is empty.</p>
<code>LabelKey</code>	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Key for the set of labels to display for these action link states: new, pending, success, failed. For example, the Approve set contains these labels: Approve,</p>

Field Name	Details
	<p>Pending, Approved, Failed. For a complete list of keys and labels, see Action Links Labels in the <i>Chatter REST API Developer Guide</i> or the <i>Apex Developer Guide</i>. If none of the label key values make sense for an action link, set this field to <code>None</code> and enter a custom label name in the <code>Label</code> field.</p>
LinkType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of action link. One of these values:</p> <ul style="list-style-type: none"> • <code>Api</code>—The action link calls a synchronous API at the action URL. Salesforce sets the status to <code>SuccessfulStatus</code> or <code>FailedStatus</code> based on the HTTP status code returned by your server. • <code>ApiAsync</code>—The action link calls an asynchronous API at the action URL. The action remains in a <code>PendingStatus</code> state until a third party makes a request to <code>/connect/action-links/actionLinkId</code> to set the status to <code>SuccessfulStatus</code> or <code>FailedStatus</code> when the asynchronous operation is complete. • <code>Download</code>—The action link downloads a file from the action URL. • <code>Ui</code>—The action link takes the user to a Web page at the action URL.
Method	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description HTTP method for the action URL. One of these values:</p> <ul style="list-style-type: none"> • <code>HttpDelete</code>—Returns HTTP 204 on success. Response body or output class is empty. • <code>HttpGet</code>—Returns HTTP 200 on success. • <code>HttpHead</code>—Returns HTTP 200 on success. Response body or output class is empty. • <code>HttpPatch</code>—Returns HTTP 200 on success or HTTP 204 if the response body or output class is empty. • <code>HttpPost</code>—Returns HTTP 201 on success or HTTP 204 if the response body or output class is empty. Exceptions are the batch posting resources and methods, which return HTTP 200 on success. • <code>HttpPut</code>—Return HTTP 200 on success or HTTP 204 if the response body or output class is empty. <p><code>Ui</code> and <code>Download</code> action links must use <code>HttpGet</code>.</p>

Field Name	Details
Position	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description An integer specifying the position of the action link template relative to other action links in the group. 0 is the first position.</p>
RequestBody	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Template for the HTTP request body sent when corresponding action links are invoked. This field can be used only for <code>Api</code> and <code>ApiAsync</code> action links. This field can contain context variables and binding variables in the form <code>{!Bindings.key}</code>.</p>
UserAlias	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description If you selected <code>CustomUser</code> or <code>CustomExcludedUser</code> for <code>UserVisibility</code>, this field is the alias for the custom user. Use the alias in a template binding to specify the custom user when an action link group is created using the template.</p>
UserVisibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Who can see the action link. This value is set per action link, not per action link group. One of these values:</p> <ul style="list-style-type: none"> • <code>Creator</code>—Only the creator of the action link can see the action link. • <code>Everyone</code>—Everyone can see the action link. • <code>EveryoneButCreator</code>—Everyone but the creator of the action link can see the action link. • <code>Manager</code>—Only the manager of the creator of the action link can see the action link. • <code>CustomUser</code>—Only the custom user can see the action link.

Field Name	Details
	<ul style="list-style-type: none"> CustomExcludedUser—Everyone but the custom user can see the action link.

Usage

Create action link templates in Setup. Use Apex classes in the `ConnectApi` namespace or Chatter REST API to instantiate action links from templates and to post feed elements with the action links.

For information about action links, see “Working with Action Links” in the *Apex Developer Guide* or the *Chatter REST API Developer Guide*.

ActivityHistory

This read-only object is displayed in a related list of closed activities—past events and closed tasks—related to an object. It includes activities for all contacts related to the object. ActivityHistory fields for phone calls are only available if your organization uses Salesforce CRM Call Center.

Supported Calls

`describeSObjects()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the related account, which is determined as follows:</p> <ul style="list-style-type: none"> The account associated with the <code>WhatId</code>, if it exists; or The account associated with the <code>WhoId</code>, if it exists; otherwise <code>null</code> <p>For information on IDs, see ID Field Type.</p>
ActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates one of the following:</p>

Field	Details
	<ul style="list-style-type: none"> The due date of a task The due date of an event if <code>IsAllDayEvent</code> is set to <code>true</code> <p>This field has a time stamp that is always set to midnight in the Universal Time Coordinated (UTC) time zone. The time stamp doesn't represent the time of the activity; don't attempt to alter it to accommodate time zone differences. Label is <code>Date</code>.</p>
ActivitySubtype	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Provides standard subtypes to facilitate creating and searching for specific activity subtypes. This field isn't updateable.</p> <p>ActivitySubtype values:</p> <ul style="list-style-type: none"> Task Email Call Event
ActivityType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents one of the following values: Call, Meeting, or Other. Label is <code>Type</code>.</p>
AlternateDetailId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of a record the activity is related to which contains more details about the activity. For example, an activity can be related to an EmailMessage record.</p>
CallDisposition	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the result of a given call, for example, "we'll call back," or "call unsuccessful." Limit is 255 characters.</p>

Field	Details
CallDurationInSeconds	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Duration of the call in seconds.</p>
CallObject	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of a call center. Limit is 255 characters.</p>
CallType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of call being answered: Inbound, Internal, or Outbound.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the PartnerNetworkConnection that shared this record with your organization. This field is available only if your organization has enabled Salesforce to Salesforce and only in API versions 28.0 and later.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the PartnerNetworkConnection that your organization shared this record with. This field is available only if your organization has enabled Salesforce to Salesforce, and only in API versions 28.0 and later. The value is always null. You can use the PartnerNetworkRecordConnection object to forward records to connections.</p>

Field	Details
Description	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description Contains a description of the event or task. Limit is 32 KB.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.</p>
DurationInMinutes	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the duration of the event or task.</p>
EndDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates the end date and time of the event or task. Available in versions 27.0 and later. This field is optional, depending on the following:</p> <ul style="list-style-type: none"> • If <code>IsAllDayEvent</code> is true, you can supply a value for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are <code>null</code>, the duration defaults to one day. • If <code>IsAllDayEvent</code> is false, a value must be supplied for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time.
IsAllDayEvent	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If the value of this field is set to <code>true</code>, then the activity is an event spanning a full day, and the <code>ActivityDate</code> defines the date of the event. If the value of this field is set to <code>false</code>, then the activity may be an event spanning less than a full day, or it may be a task. Label is <code>All-Day Event</code>.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a task is closed; value is always <code>true</code>. This field is set indirectly by setting the <code>Status</code> field on the task—each picklist value has a corresponding <code>IsClosed</code> value. Label is <code>Closed</code>.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the activity has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is <code>Deleted</code>.</p>
IsHighPriority	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates a high-priority task. This field is derived from the <code>Priority</code> field.</p>
IsOnlineMeeting	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the activity represents an online meeting (<code>true</code>) or not (<code>false</code>).</p> <p> Note: This field is not available in API version 16.0 or later.</p>

Field	Details
IsReminderSet	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a reminder is set for an activity (<code>true</code>) or not (<code>false</code>).</p>
IsTask	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If the value of this field is set to <code>true</code>, then the activity is a task. If the value is set to <code>false</code>, then the activity is an event. Label is <code>Task</code>.</p>
IsVisibleInSelfService	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description If the value of this field is set to <code>true</code>, then the activity can be viewed in the self-service portal. Label is <code>Visible in Self-Service</code>.</p>
Location	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the activity is an event, then this field contains the location of the event. If the activity is a task, then the value is <code>null</code>.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the user who owns the activity.</p>
PrimaryAccountId	<p>Type reference</p>

Field	Details
	Properties
	Filter, Group, Nillable, Sort
	Description
	Contains the AccountId value from the activity record. Available in API versions 30.0 and later to organizations that use Shared Activities.
PrimaryWhoId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	Contains the WhoId value from the activity record. Available in API versions 30.0 and later to organizations that have enabled Shared Activities.
Priority	Type
	picklist
	Properties
	Defaulted on create, Filter, Group, Nillable, Sort
	Description
	Indicates the priority of a task, such as high, normal, or low.
ReminderDateTime	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	Represents the time when the reminder is scheduled to fire, if IsReminderSet is set to true. If IsReminderSet is set to false, then the user may have deselected the reminder checkbox in the Salesforce user interface, or the reminder has already fired at the time indicated by the value.
Status	Type
	picklist
	Properties
	Defaulted on create, Filter, Group, Nillable, Sort
	Description
	Indicates the current status of a task, such as in progress or complete. Each predefined status field sets a value for IsClosed. To obtain picklist values, query TaskStatus.
Subject	Type
	combobox

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Contains the subject of the task or event.</p>
WhatId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The WhatId represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. WhatIds are polymorphic. Polymorphic means a WhatId is equivalent to the ID of a related object. The label is Related To ID.</p>
WhoId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The WhoId represents a human such as a lead or a contact. Whoids are polymorphic. Polymorphic means a WhoId is equivalent to a contact's ID or a lead's ID. The label is Name ID. If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the WhoId field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is Name ID. If your organization uses Shared Activities, when you query activities in API version 30.0 or later, the returned value of the WhoId field matches the value in the queried object, not necessarily in the activity record itself.</p>

Usage

Query activities that are related to an object

1. Optionally, issue a describe call against the object whose activities you wish to query, to get a suggestion of the correct SOQL to use.
2. Issue a SOQL relationship query with a main clause that references the object, and an inner clause that references the activity history; for example:

```
SELECT
  (SELECT ActivityDate, Description
   FROM ActivityHistories)
```

```
FROM Account
WHERE Name Like 'XYZ%'
```

The user interface enforces sharing rules, filtering out related-list items that a user doesn't have permission to see.

The following constraints on users who don't have the "View All Data" permission help prevent performance issues.

- In the main clause of the relationship query, you can reference only one record. For example, you can't filter on all records where the account name starts with "A." Instead, you must reference a single account record.

```
SELECT
  (SELECT ActivityDate, Description
   FROM ActivityHistory
   ORDER BY ActivityDate DESC NULLS LAST, LastModifiedDate DESC
   LIMIT 500)
  FROM Account
  WHERE Name = 'Acme'
  LIMIT 1
```

- In the inner clause of the query, you can't use WHERE.
- In the inner clause of the query, you must specify a limit of 500 or fewer on the number of rows that are returned in the list.
- In the inner clause of the query, you must sort on ActivityDate in descending order and LastModifiedDate in descending order. You can optionally display nulls last. For example: ORDER BY ActivityDate DESC NULLS LAST, LastModifiedDate DESC.

SEE ALSO:

[Task](#)

AdditionalNumber

Represents an optional additional number for a call center. This additional number is visible in the call center's phone directory.

Supported Calls

```
create(), delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve(),
undelete(), update(), upsert()
```

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CallCenterId	Type
	reference

Field	Details
	<p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description System field that contains the ID of the user who created the call center associated with this additional number. If value is null, this additional number is displayed in every call center's phone directory.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Description of the additional number, such as Conference Room B. Limit: 255 characters.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the additional number. Limit: 80 characters.</p>
Phone	<p>Type phone</p> <p>Properties Create, Filter, Nullable, Group, Sort, Update</p> <p>Description The phone number that corresponds to this additional number.</p>

Usage

Create an additional number for a call center directory. Use this object if the number is not easily categorized as a User, Contact, Lead, Account, or the other object. Examples include phone queues or conference rooms.

AgentWork

Represents a work assignment that's been routed to an agent. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), query(), getDeleted(), getUpdated(), retrieve(), undelete()`

Fields

Field	Details
AcceptDatetime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was accepted.</p>
ActiveTime	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time an agent actively worked on the work item. Tracks when the item is open and in focus in the agent's console.</p>
AgentCapacityWhenDeclined	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The agent's capacity when declining work, either explicitly or through push timeout.</p>
AssignedDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was assigned to an agent.</p>
CancelDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was canceled.</p>

Field	Details
CapacityPercentage	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel. When combined work items reach 100% for an agent, the agent won't receive new work items until there is enough open capacity for more work. For example, if you give phone calls a capacity percentage of 100, an agent on a call doesn't receive new work items until the call ends.</p>
CapacityWeight	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel. For example, if cases are assigned a capacity weight of 2, an agent with a capacity of 6 can accept up to 3 cases before the agent is at capacity and can't receive new work items.</p>
CloseDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was closed.</p>
CreatedBy	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the User who created this record.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on createFilter, Sort</p> <p>Description Date and time when this record was created.</p>

Field	Details
DeclineDatetime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when this record was declined by an agent.</p>
DeclineReason	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The provided reason for why an agent declined the work request.</p>
HandleTime	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time an agent had the work item open. Calculated by Close Time – Accepted Time.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, idLookup, Sort</p> <p>Description The ID of the AgentWork object.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
LastModifiedById	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description The ID of the user who last modified this record.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description An automatically generated ID number that identifies the record.</p>
OriginalQueueId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the queue that the work assignment was originally routed to.</p>
PushTimeout	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of seconds set for push timeout. 0 is returned when push timeout isn't enabled. Available in API version 36.0 and later.</p>
PushTimeoutDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the push timeout event occurred. Available in API version 36.0 and later.</p>
RequestDateTime	<p>Type dateTime</p>

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work was requested.</p>
ServiceChannelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the service channel that's associated with the work assignment.</p>
SpeedToAnswer	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time between when the work was requested and when an agent accepted it.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The working status of the work item. Valid values are:</p> <ul style="list-style-type: none"> • Assigned – The item is assigned to the agent but hasn't been opened. • Opened – The item was opened by the agent. • Unavailable – The item was assigned to the agent but the agent became unavailable (went offline or lost connection). • Declined – The item was assigned to the agent but the agent explicitly declined it. • DeclinedOnPushTimeout – The item was declined because push time-out is enabled and the item request timed out with the agent. • Closed – The item is closed. • Canceled – The item no longer needs to be routed. For example: a chat visitor cancels their Omni-Channel routed chat request before it reaches an agent.
SystemModstamp	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when a user or automated process (such as a trigger) last modified this record.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user that the work item was assigned to.</p>
WorkItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the object that's routed to the agent through Omni-Channel.</p>

Usage

`AgentWork` records can only be deleted if they have the status Closed, Declined, or Unavailable. They can't be deleted if their status is Assigned or Opened because they're active in Omni-Channel.

`AgentWork` records have the status Assigned when they're created. Once created, the record is automatically pushed to the assigned agent.

While the metadata for `AgentWork` indicates support for `upsert()` and `update()`, these calls aren't used with `AgentWork` because none of its fields can be updated.

AllowedEmailDomain

Represents an allowed email domain for users in your organization. You can define a whitelist to restrict the email domains allowed in a user's `Email` field. This object is available in API version 29.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

You must have the "Manage Internal Users" user permission to use this object.

 **Note:** If you don't see this object, contact your Salesforce representative to enable it.

Fields

Field	Details
Domain	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description An allowed email domain for users.</p>

Announcement

Represents a Chatter group announcement. This object is available in API version 30.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
ExpirationDate	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description Required. The date on which the announcement expires. Announcements display on the group UI until 11:59 p.m. local time on the selected date.</p>
FeedItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. The ID of the FeedItem that contains the content of the announcement. Announcements are stored as text posts.</p>
ParentId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the parent CollaborationGroup that the announcement belongs to. An announcement can belong only to a single Chatter group.</p>
SendEmails	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Set to <code>true</code> to email all group members when an announcement is posted to the group. The default is <code>false</code>. This requires the user to have the “Send announcement on email” permission.</p> <p>This field is available in API version 36.0 and later.</p> <p> Note: This field is currently available to select customers through a pilot program. To be nominated to join this pilot program, contact Salesforce. Additional terms and conditions may apply to participate in the pilot program. Please note that pilot programs are subject to change, and as such, we cannot guarantee acceptance into this pilot program or a particular time frame in which this feature can be enabled. Any unreleased services or features referenced in this document, press releases, or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make their purchase decisions based upon features that are currently available.</p>

Usage

Group owners, managers, and users with the “Modify All Data” permission can use the Announcement object to create, edit, and delete group announcements. Creating a group announcement is a three-step process.

1. Use the FeedItem object to create a text post with the announcement’s content. Use the CollaborationGroup record you want to post the announcement to as the parent of this feed item.
2. Next, use the feed item ID and an expiration date to create the announcement record.
3. Finally, update the `AnnouncementId` field in the CollaborationGroup record with the ID of the announcement you created.

To delete the group announcement, simply delete the `AnnouncementId` value in the CollaborationGroup record. To restore a group announcement, update the `AnnouncementId` field for a group with the announcement’s ID. The expiration date for the announcement should be in the future and the feed item used to create the announcement should be parented by the same group.

ApexClass

Represents an Apex class.



Note: Although Apex classes and triggers have the Create and Update field properties, a runtime exception occurs if you try to create or update them using the API. Instead, use the Force.com Migration Tool, the Salesforce user interface, or the Force.com IDE to create or update Apex classes or triggers.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
ApiVersion	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The API version for this class. Every class has an API version specified at creation.</p>
Body	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The Apex class definition. Limit: 1 million characters.</p>
bodyCrc	<p>Type double</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description The CRC (cyclic redundancy check) of the class or trigger file.</p>
IsValid	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Indicates whether any dependent metadata has changed since the class was last compiled (<code>true</code>) or not (<code>false</code>).</p>
LengthWithoutComments	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Length of the class without comments.</p>
Name	<p>Type string</p> <p>Properties Group, Sort, Create, Filter, Update</p> <p>Description Name of the class. Limit: 255 characters</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Status	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
	<p>Description The current status of the Apex class. The following string values are valid:</p> <ul style="list-style-type: none"> • Active—The class is active. • Deleted—The class is marked for deletion. This is useful for managed packages, because it allows a class to be deleted when a managed package is updated. <p> Note: The ApexTrigger <code>status</code> field includes an <code>Inactive</code> option, but it is only supported for ApexTrigger. For more information, see the Metadata API Developer Guide.</p>

SEE ALSO:

[ApexTrigger](#)

[Developer Guide: Apex Developer Guide](#)

ApexComponent

Represents a definition for a custom component that can be used in a Visualforce page alongside standard components such as `<apex:relatedList>` and `<apex:dataTable>`.

Represents a definition for a custom component that can be used in a Visualforce page alongside standard components such as `<apex:relatedList>` and `<apex:dataTable>`. For information, see the [Visualforce Developers Guide](#).

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
ApiVersion	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The API version for this custom component. Every custom component has an API version specified at creation. If the API version is less than 15.0 and <code>ApiVersion</code> is not specified, <code>ApiVersion</code> defaults to 15.0.</p>

Field	Details
ControllerKey	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The identifier for the controller associated with this custom component:</p> <ul style="list-style-type: none"> • If the ControllerType parameter is set to Standard or StandardSet, this value is the name of the sObject that defines the controller. • If the ControllerType parameter is set to Custom, this value is the name of the Apex class that defines the controller.
ControllerType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of controller associated with this Visualforce custom component. Possible values include:</p> <ul style="list-style-type: none"> • Not Specified, for custom components defined without a value for the controller attribute on the <code><apex:component></code> tag • Standard, a value that can't be used with custom components or errors may occur • StandardSet, a value that can't be used with custom components or errors may occur • Custom, for components that have a value for the controller attribute on the <code><apex:component></code> tag
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the Visualforce custom component.</p>
Markup	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The Visualforce markup, HTML, Javascript, and any other Web-enabled code that defines the content of the custom component.</p>

Field	Details
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text used to identify the Visualforce custom component in the Setup area of Salesforce. The Label for this field is Label.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of this Visualforce custom component.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

Use custom components to encapsulate a common design pattern and then reuse that pattern several times in one or more Visualforce pages. All users who can view Visualforce pages can view custom components, but the “Customize Application” permission is required to create or update custom components.

SEE ALSO:

[ApexPage](#)
[StaticResource](#)
[Developer Guide: Visualforce Developer Guide](#)

ApexLog

Represents a debug log, containing information about a transaction, including information about Apex, Visualforce, workflow and validation rules. This object is available in API version 19.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
Application	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description This value depends on the client type that triggered the log.</p> <ul style="list-style-type: none">For API clients, this value is the client IDFor browser clients, this value is <code>Browser</code>
DurationMilliseconds	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Duration of the transaction in milliseconds.</p>
Location	<p>Type picklist</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies the location of the origin of the log. Values are:</p> <ul style="list-style-type: none"> • Monitoring—log is generated as part of debug log monitoring. These types of logs are maintained until the user or the system overwrites them. • SystemLog—log is generated from the Developer Console. These types of logs are only maintained for 60 minutes or until the user clears them.
LogLength	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Length of the log in bytes.</p>
LogUserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user whose actions triggered the debug log.</p>
Operation	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Name of the operation that triggered the debug log, such as APEXSOAP, Apex Sharing Recalculation, and so on.</p>
Request	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Request type. Values are:</p> <ul style="list-style-type: none"> • API—request came from the API • Application—request came from the Salesforce user interface

Field	Details
StartTime	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description Start time of the transaction.</p>
Status	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Status of the transaction. This value is either <code>Success</code>, or the text of an unhandled Apex exception.</p>

Usage

You can read information about this object, as well as delete it, but you can't update or insert it.

SEE ALSO:

[ApexClass](#)

[ApexTrigger](#)

[Developer Guide: Apex Developer Guide](#)

ApexPage

Represents a single Visualforce page.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
ApiVersion	<p>Type double</p>

Field	Details
	<p>Properties Create, Filter, Sort, Update</p> <p>Description The API version for this page. Every page has an API version specified at creation. If the API version is less than 15.0 and <code>ApiVersion</code> is not specified, <code>ApiVersion</code> defaults to 15.0.</p>
ControllerKey	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The identifier for the controller associated with this page:</p> <ul style="list-style-type: none"> • If the <code>ControllerType</code> parameter is set to <code>Standard</code> or <code>StandardSet</code>, this value is the name of the sObject that defines the controller. • If the <code>ControllerType</code> parameter is set to <code>Custom</code>, this value is the name of the Apex class that defines the controller.
ControllerType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of controller associated with this Visualforce page. Possible values include:</p> <ul style="list-style-type: none"> • <code>Not Specified</code>, for pages defined with neither a <code>standardController</code> nor a <code>controller</code> attribute on the <code><apex:page></code> tag • <code>Standard</code>, for pages defined with the <code>standardController</code> attribute on the <code><apex:page></code> tag • <code>StandardSet</code>, for pages defined using the <code>standardController</code> and <code>recordSetVar</code> attribute on the <code><apex:page></code> tag • <code>Custom</code>, for pages defined with the <code>controller</code> attribute on the <code><apex:page></code> tag
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the Visualforce page.</p>
isAvailableInTouch	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates if Visualforce tabs associated with the Visualforce page can be used in the Salesforce1 app (<code>true</code>) or not (<code>false</code>). (Use of this field for Salesforce Touch is deprecated.) This field is available in API version 27.0 and later. Standard object tabs that are overridden with a Visualforce page aren't supported in Salesforce1, even if you set this field for the page. The default Salesforce1 page for the object is displayed instead of the Visualforce page.</p>
IsConfirmationTokenRequired	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether GET requests for the page require a CSRF confirmation token (<code>true</code>) or not (<code>false</code>). This field is available in API version 28.0 and later. If you change this field's value from <code>false</code> to <code>true</code>, links to the page require a CSRF token to be added to them, or the page will be inaccessible.</p>
Markup	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The Visualforce markup, HTML, Javascript, and any other Web-enabled code that defines the content of the page.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text used to identify the Visualforce page in the Setup area of Salesforce. The Label is Label.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of this Visualforce page.</p>

Field	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

Use Visualforce pages to add custom content that extends the base Salesforce application functionality. All users in Visualforce-enabled organizations can view Visualforce pages, but the “Customize Application” permission is required to create or update them.

SEE ALSO:

[ApexComponent](#)

[StaticResource](#)

[Developer Guide: Visualforce Developer Guide](#)

ApexTestQueueItem

Represents a single Apex class in the Apex job queue. This object is available in API version 23.0 and later.

This object is available in API version 23.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Description
ApexClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The Apex class whose tests are to be executed.</p>
ExtendedStatus	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The pass rate of the test run. For example: "(4/6)". This means that four out of a total of six tests passed. If the class fails to execute, this field contains the cause of the failure.</p>
ParentJobId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Points to the AsyncApexJob that represents the entire test run. If you insert multiple Apex test queue items in a single bulk operation, the queue items will share the same parent job. This means that a test run can consist of the execution of the tests of several classes if all the test queue items are inserted in the same bulk operation.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status of the job. Valid values are:</p> <ul style="list-style-type: none"> • Holding¹ • Queued • Preparing • Processing • Aborted

Field Name	Description
	<ul style="list-style-type: none"> Completed Failed
¹ This status applies to batch jobs in the Apex flex queue.	
TestRunResultID	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the associated ApexTestRunResult object.</p>

Usage

Insert an `ApexTestQueueItem` object to place its corresponding Apex class in the Apex job queue for execution. The Apex job executes the test methods in the class.

To abort a class that is in the Apex job queue, perform an update operation on the `ApexTestQueueItem` object and set its `Status` field to `Aborted`.

If you insert multiple Apex test queue items in a single bulk operation, the queue items will share the same parent job. This means that a test run can consist of the execution of the tests of several classes if all the test queue items are inserted in the same bulk operation.

ApexTestResult

Represents the result of an Apex test method execution. This object is available in API version 23.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
ApexClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The Apex class whose test methods were executed.</p>

Field Name	Details
ApexLogId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Points to the ApexLog for this test method execution if debug logging is enabled; otherwise, null.</p>
ApexTestRunResultId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the ApexTestRunResult that represents the entire test run.</p>
AsyncApexJobId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Points to the AsyncApexJob that represents the entire test run. This field points to the same object as ApexTestQueueItem.ParentJobId.</p>
Message	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The exception error message if a test failure occurs; otherwise, null.</p>
MethodName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The test method name.</p>
Outcome	<p>Type picklist</p>

Field Name	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update
	Description The result of the test method execution. Can be one of these values:
	<ul style="list-style-type: none"> ● Pass ● Fail ● CompileFail ● Skip
QueueItemId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Points to the ApexTestQueueItem which is the class that this test method is part of.
RunTime	Type int
	Properties Create, Filter, Nillable, Sort, Update
	Description The time it took the test method to run, in seconds.
StackTrace	Type string
	Properties Create, Filter, Nillable, Sort, Update
	Description The Apex stack trace if the test failed; otherwise, null.
TestTimestamp	Type dateTime
	Properties Create, Filter, Sort, Update
	Description The start time of the test method.

Usage

You can query the fields of the `ApexTestResult` record that corresponds to a test method executed as part of an Apex class execution.

Each test method execution is represented by a single `ApexTestResult` record. For example, if an Apex test class contains six test methods, six `ApexTestResult` records are created. These records are in addition to the `ApexTestQueueItem` record that represents the Apex class.

Each `ApexTestResult` record has an associated [ApexTestResultLimits](#) on page 283 record, which captures the Apex limits used during execution of the test method.

ApexTestResultLimits

Captures the Apex test limits used for a particular test method execution. An instance of this object is associated with each `ApexTestResult` record. This object is available in API version 37.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
<code>ApexTestId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the associated ApexTestResult object.</p>
<code>AsyncCalls</code>	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of asynchronous calls made during the test run.</p>
<code>Callouts</code>	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of callouts made during the test run.</p>

Field Name	Details
Cpu	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The amount of CPU used during the test run, in milliseconds.</p>
Dml	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of DML statements made during the test run.</p>
DmlRows	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of rows accessed by DML statements during the test run.</p>
Email	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of email invocations made during the test run.</p>
LimitContext	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates whether the test run was synchronous or asynchronous.</p>
LimitExceptions	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
Description	
Indicates whether your org has any limits that differ from the default limits.	
MobilePush	
Type	
int	
Properties	
Create, Filter, Group, Sort, Update	
Description	
The number of mobile push calls made during the test run.	
QueryRows	
Type	
int	
Properties	
Create, Filter, Group, Sort, Update	
Description	
The number of rows queried during the test run.	
Soql	
Type	
int	
Properties	
Create, Filter, Group, Sort, Update	
Description	
The number of SOQL queries made during the test run.	
Sosl	
Type	
int	
Properties	
Create, Filter, Group, Sort, Update	
Description	
The number of SOSL queries made during the test run.	

Usage

The `ApexTestResultLimits` object is populated for each test method execution, and it captures the limits used between the `Test.startTest()` and `Test.stopTest()` methods. If `startTest()` and `stopTest()` aren't called, limits usage is not captured. Note the following:

- The associated test method must be run asynchronously.
- Limits for asynchronous Apex operations (batch, scheduled, future, and queueable) that are called within test methods are not captured.
- Limits are captured only for the default namespace.

ApexTestRunResult

Contains summary information about all the test methods that were run in a particular Apex job. This object is available in API version 37.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
AsyncApexJobId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The parent Apex job ID for the result.</p>
ClassesCompleted	<p>Type int</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The total number of classes executed during the test run.</p>
ClassesEnqueued	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The total number of classes enqueued during the test run.</p>
EndTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description The time at which the test run ended.</p>

Field Name	Details
IsAllTests	Type boolean
	Properties Create, Filter, Group, Sort, Update
	Description
	Indicates whether all Apex test classes were run.
JobName	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description
	Reserved for future use.
MethodsCompleted	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description
	The total number of methods completed during the test run. This value is updated after each class is run.
MethodsEnqueued	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description
	The total number of methods enqueued for the test run. This value is initialized before the test runs.
MethodsFailed	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description
	The total number of methods that failed during this test run. This value is updated after each class is run.
Source	Type string

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The source of the test run, such as the Developer Console.</p>
StartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The time at which the test run started.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The status of the test run. Values include:</p> <ul style="list-style-type: none">● Queued● Processing● Aborted● Completed● Failed
TestTime	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The time it took the test to run, in seconds.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user who ran the test run.</p>

ApexTestSuite

Represents a suite of Apex classes to include in a test run. A TestSuiteMembership object associates each class with the suite. This object is available in API version 36.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Description
TestSuiteName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Unique, Update</p> <p>Description The name of the Apex test suite. This label appears in the user interface. This value is case-sensitive and must be unique.</p>

Usage

Insert a TestSuiteMembership object using an API call to associate an Apex class with an ApexTestSuite object. (ApexTestSuite and TestSuiteMembership aren't editable through Apex DML.) To remove the class from the test suite, delete the TestSuiteMembership object. If you delete an Apex test class or test suite, all TestSuiteMembership objects that contain that class or suite are deleted.

The following SOQL query returns the membership object that relates this Apex class to this test suite.

```
SELECT Id FROM TestSuiteMembership WHERE ApexClassId = '01pD000000Fhy9IAC'  
AND ApexTestSuiteId = '05FD00000004CDBMA2'
```

SEE ALSO:

[TestSuiteMembership](#)

ApexTrigger

Represents an Apex trigger.

 **Note:** Although Apex classes and triggers have the Create and Update field properties, a runtime exception occurs if you try to create or update them using the API. Instead, use the Force.com Migration Tool, the Salesforce user interface, or the Force.com IDE to create or update Apex classes or triggers.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Fields

Field	Details
ApiVersion	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The API version for this trigger. Every trigger has an API version specified at creation.</p>
Body	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description The Apex trigger definition. Limit: 1 million characters.</p>
bodyCrc	<p>Type double</p> <p>Properties Create, Defaulted on create, Filter, Nullable, Sort, Update</p> <p>Description The CRC (cyclic redundancy check) of the class or trigger file.</p>
IsValid	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether any dependent metadata has changed since the trigger was last compiled (<code>true</code>) or not (<code>false</code>).</p>
LengthWithoutComments	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Length of the trigger without comments</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the trigger. Limit: 255 characters</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The current status of the Apex trigger. The following string values are valid:</p> <ul style="list-style-type: none"> • Active—The trigger is active. • Inactive—The trigger is inactive, but not deleted. • Deleted—The trigger is marked for deletion. This is useful for managed packages, because it allows a class to be deleted when a managed package is updated.

Field	Details
	<p> Note: <code>Inactive</code> is not valid for ApexClass. For more information, see the Metadata API Developer Guide.</p>
TableEnumOrId	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies the object associated with the trigger, such as Account or Contact.</p>
UsageAfterDelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an <code>after delete</code> trigger (<code>true</code>) or not (<code>false</code>).</p>
UsageAfterInsert	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an <code>after insert</code> trigger (<code>true</code>) or not (<code>false</code>).</p>
UsageAfterUndelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an <code>after undelete</code> trigger (<code>true</code>) or not (<code>false</code>).</p>
UsageAfterUpdate	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an <code>after update</code> trigger (<code>true</code>) or not (<code>false</code>).</p>
UsageBeforeDelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
	Description
	Specifies whether the trigger is a <code>before delete</code> trigger (<code>true</code>) or not (<code>false</code>).
UsageBeforeInsert	Type boolean
	Properties
	Create, Filter, Update
	Description
	Specifies whether the trigger is a <code>before insert</code> trigger (<code>true</code>) or not (<code>false</code>).
UsageBeforeUpdate	Type boolean
	Properties
	Create, Filter, Update
	Description
	Specifies whether the trigger is a <code>before update</code> trigger (<code>true</code>) or not (<code>false</code>).
UsageIsBulk	Type boolean
	Properties
	Create, Filter, Update
	Description
	Specifies whether the trigger is defined as a bulk trigger (<code>true</code>) or not (<code>false</code>).
 Note: This field is not used for Apex triggers saved using Salesforce API version 10.0 or higher; all triggers starting with that version are automatically considered bulk, and this field will always return <code>true</code> .	

SEE ALSO:

[ApexClass](#)

[Developer Guide: Apex Developer Guide](#)

AppMenuItem

Represents the organization's default settings for items in the Force.com app menu or App Launcher.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApplicationId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 15-character ID for the menu item.</p>
CanvasAccessMethod	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The access method for the canvas app. Values can be:</p> <ul style="list-style-type: none"> • OAuth Webflow (GET) • Signed Request (POST)
CanvasEnabled	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates if the app menu item is a canvas app (<code>true</code>) or not (<code>false</code>).</p>
CanvasOptions	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the options enabled for a canvas connected app. The options are:</p> <ul style="list-style-type: none"> • PersonalEnabled—The app is enabled as a canvas personal app. • HideHeader—The publisher header, which contains the “What are you working on?” text, is hidden. • HideShare—The publisher Share button is hidden. <p>This field is available in API version 34.0 and later.</p>
CanvasReferenceId	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The canvas app unique identifier.</p>
CanvasSelectedLocations	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The selected locations for the canvas app which define where the canvas app can appear in the user interface. For example:</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Chatter, ChatterFeed, Publisher, ServiceDesk</div>
CanvasUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL of the canvas app.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A description of this menu item.</p>
IconUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The icon for the menu item's application.</p>
InfoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL for more information about the application.</p>

Field	Details
IsAccessible	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the current user is authorized to use the app.</p>
IsUsingAdminAuthorization	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the app is pre-authorized for certain users by the administrator.</p>
IsVisible	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the app is visible to users of the organization, by default.</p>
Label	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The app's name.</p>
LogoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The logo for the menu item's application. The default is the initials of the <code>Label</code> value.</p>
MobileStartUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The location mobile users are directed to after they've authenticated. This field is only used with connected apps.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The API name of the item.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The index value that controls where this item appears in the menu. For example, a menu item with a sort order of 5 appears between items with sort order values of 3 and 9.</p>
StartUrl	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description For a connected app, the location users are directed to after they've authenticated. Otherwise, the application's default start page.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of application represented by this item. The types are:</p> <ul style="list-style-type: none"> • ConnectedApplication • ServiceProvider • TabSet
UserSortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The index value that represents where the user set this item in the menu (or App Launcher). For example, an item with a sort order value of 5 appears between items with sort order values of 3 and 9. This value is separate from SortOrder so you can create logic incorporating both values. For example, if you want the user-sorted items to appear first, followed by the organization order for the rest, use:</p> <pre>SELECT ApplicationId, SortOrder, UserSortOrder FROM AppMenuItem order by userSortOrder NULLS LAST, sortOrder NULLS LAST</pre>

Usage

Use this read-only object to view an entry in the Force.com app menu or the App Launcher. You can create a SOQL query to retrieve all items, even items the user does not see from the user interface.

There are many ways you can use AppMenuItem. Here are some examples:

- Build your own App Launcher or app menu in Salesforce. Create a custom page showing all the apps you have access to and that lets you run them using single sign-on.
- Build your own App Launcher or app menu on a tablet or mobile app. You can have your own app for launching applications on various mobile devices.

- Build an app launcher into your company's intranet. There's no need to have it run on Salesforce because there's a Web service API to let you build an app launcher.

 **Tip:** To get metadata information about apps and their tabs, use the Apex `Schema.describeTabs()` method, the REST API `/vXX.X/tabs/` resource, or the SOAP API `describeTabs()` call.

Approval

Represents an approval request for a Contract.

 **Note:** This object is read-only and is specific to approvals on the Contract object. It isn't equal to or involved in the approval processes represented by the ProcessInstance, which is more powerful.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ApproveComment	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Text entered by the user when they approved or rejected this approval request. Required. Limit: 4,000 characters.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description Required. ID of the User being asked to approve or reject the approval request. Must be a valid User ID. Required.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. ID of the Contract associated with this approval request. Must be a valid contract ID.</p>
RequestComment	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Text entered by the User who created the approval request. Optional. This field can't be updated after the Approval has been created. Limit: 4,000 characters.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Required. Status of this approval request. One of the following picklist values:</p> <ul style="list-style-type: none"> • Pending—Specified only when the Approval request is created (<code>create ()</code> call) • Approved—Specified only when the Approval request is approved (<code>update ()</code> call) • Rejected—Specified when the Approval request is rejected (<code>update ()</code> call) or when it is created (<code>create ()</code> call) and immediately rejected for archival/historical purposes.

Usage

This object allows client applications to programmatically handle approval requests for a Contract. Initially, to request a Contract approval, a client application might create a new Approval request record, specifying the `ParentId`, `OwnerId` (user approving or rejecting the request), `Status` (Pending), and (optionally) `RequestComment` fields. Note that when a client application creates the first approval request, if the value of the Contract `Status` field is Draft, then the Approval `Status` for this record is automatically changed to In Approval Process (see `ContractStatus` for more information).

A client application might subsequently update an existing Approval request, specifying the `Status` (Approved or `Rejected`) and an `ApproveComment` (required); the `RequestComment` field can't be updated. Updating an Approval record (either to approve or reject) requires the client application to be logged in with "Approve Contract" permission. To update an Approval request, its `Status`

must be Pending—a client application can't update an Approval that has already been Approved or Rejected. To re-submit an approval request for a given Contract, a client application must create a new, separate Approval record and repeat the approval process.

Once a Contract has been approved (not rejected), the Contract `LastApprovedDate` field is automatically updated, however the Contract `Status` field isn't updated, it keeps the value `InApproval`.

An approved Contract must be activated explicitly. Client applications can activate a Contract by setting the value in its `Status` field to `Activated`, or a User can activate a Contract via the Salesforce user interface.

A Contract can have multiple approval requests in various states (Pending, Approved, and Rejected). In addition, one User can have multiple approval requests associated with the same Contract.

Client applications can't explicitly deleteApproval records. Approval records are deleted automatically if the parent Contract is deleted.

SEE ALSO:

[Object Basics](#)

Article Type__DataCategorySelection

A data category selection represents a data category that classifies an article. This object is available in API version 19.0 and later.

This object can be used to associate an article with data categories from a data category group or to query the category selections for an article.

The object name is variable and has a syntax of `Article Type__DataCategorySelection`, where `Article Type` is the Object Name for the article type associated with the article. For example, `Offer__DataCategorySelection` represents the association between the `Offer` article type and its data categories. Every article is associated with an article type.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `getDeleted()`, `retrieve()`

Special Access Rules

Users can only access, create or delete data category selection visible to their role, permission set, or profile. If a user has partial visibility on an article's categorization, only the visible categories are returned.

Fields

Field Name	Details
<code>DataCategoryGroupName</code>	<p>Type DataCategoryGroupReference</p> <p>Properties Create</p> <p>Description Unique name of the data category group which has categories associated with the article.</p>

Field Name	Details
DataCategoryName	<p>Type DataCategoryGroupReference</p> <p>Properties Create</p> <p>Description Unique name of the data category associated with the article.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the article associated with the data category selection.</p>

Usage

Every article in Salesforce Knowledge can be categorized. A data category selection represents a category that has been selected to classify an article. You can use the `Article Type__DataCategorySelection` object to query and manage article categorization in your organization. Client applications can create a categorization for an article with a Draft status. They can also delete and query article categorizations.

 **Note:** When using `Article Type__DataCategorySelection` to classify an article, you can't select both a category (for example USA) and one of its descendants (California) or ascendant categories (North America). In this case, only the first category is selected.

Answer communities use `QuestionDataCategorySelection` to classify questions.

SOQL Sample

The following SOQL query returns the data category selections used to classify the article whose ID is `ka0D000000005ApIAI`.

```
SELECT Id, DataCategoryName, ParentId
      FROM Offer__DataCategorySelection WHERE ParentId='ka0D000000005ApIAI'
```

This clause only returns category unique names. To retrieve category labels use the following clause:

```
SELECT Id,toLabel(DataCategoryName), ParentId
      FROM Offer__DataCategorySelection WHERE ParentId='ka0D000000005ApIAI'
```

 **Tip:** You can also use relationship queries to retrieve categorizations from an article type.

SEE ALSO:

[QuestionDataCategorySelection](#)

[Data Categories](#)

[Articles](#)

Article Type__Feed

Represents a single feed item in the feed displayed on the detail page for an article. This object is available in API version 20.0 and later.

An article feed shows recent changes to an article record for any fields that are tracked in feeds, and posts and comments about the article. It is a useful way to stay up-to-date with changes made to articles in Salesforce Knowledge. This object is available in API version 20.0 and later. The object name is variable and uses a *Article Type__Feed* syntax, where *Article Type* is the Object Name for the article type associated with the article. For example, *Offer__Feed* represents a feed item on an article based on the *Offer* article type.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of <i>Article Type__Feed</i>. Required when <i>Type</i> is <i>TextPost</i>. Optional when <i>Type</i> is <i>ContentPost</i> or <i>LinkPost</i>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ContentData	<p>Type base64</p>

Field	Details
	<p>Properties Nillable</p> <p>Description Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDate DESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags: <ul style="list-style-type: none"> • <p> </p>

Field	Details
	<p>Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p>Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the article that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item: <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence. </p>

Field	Details
	<ul style="list-style-type: none">• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RipplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• CaseCommentPost—generated event when a user adds a case comment for a case object• EmailMessageEvent—generated event when an email related to a case object is sent or received• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• ChangeStatusPost—generated event when a user changes the status of a case• AttachArticleEvent—generated event when a user attaches an article to a case <p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p>

Usage

Use this object to track changes for an article. You can only delete a feed that you created, or if you have the “Modify All Data” permission or “Modify All” permission on the KnowledgeArticle object.

SEE ALSO:

[KnowledgeArticle](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

Asset

Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased and installed.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description (Required) ID of the Account associated with this asset. Must be a valid account ID. Required if ContactId is not specified.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Required if AccountId is not specified. ID of the Contact associated with this asset. Must be a valid contact ID that has an account parent (but does not need to match the asset’s AccountId).</p>
Description	<p>Type textarea</p>

Field	Details
	<p>Properties Create, Nillable, Update</p> <p>Description Description of the asset.</p>
InstallDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date when the asset was installed.</p>
IsCompetitorProduct	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this Asset represents a product sold by a competitor (<code>true</code>) or not (<code>false</code>). Default value is <code>false</code>. Its UI label is Competitor Asset.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time that the asset was last modified. Its UI label is Last Modified Date.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time that the asset was last viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description (Required) Name of the asset. Label is Asset Name.</p>

Field	Details
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The asset's owner. By default, the asset owner is the user who created the asset record. As of Spring '15, the <code>OwnerId</code> field is used for access control when the Asset object's sharing setting is Private, Public Read-Only, or Public Read/Write. If the Asset object's sharing setting is Controlled by Parent, the <code>OwnerId</code> field has no effect on access control, because the parent account's settings control access to the asset. Its UI label is Asset Owner.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The asset's parent asset. Its UI label is Parent Asset.</p>
Price	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Price paid for this asset.</p>
Product2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description (Optional) ID of the Product2 associated with this asset. Must be a valid Product2 ID. Its UI label is Product.</p>
PurchaseDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date on which this asset was purchased.</p>
Quantity	<p>Type double</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Quantity purchased or installed.</p>
RootAssetId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The top-level asset in an asset hierarchy. Depending on where an asset lies in the hierarchy, its root could be the same as its parent. Its UI label is Root Asset.</p>
SerialNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Serial number for this asset.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Customizable picklist of values. The default picklist includes the following values:</p> <ul style="list-style-type: none"> • Purchased • Shipped • Installed • Registered • Obsolete
UsageEndDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date when usage for this asset ends or expires.</p>

Usage

Use this object to track assets previously sold into customer accounts. With asset tracking, a client application can quickly determine which products were previously sold or are currently installed at a specific account. You can also create hierarchies of assets using the `Parent Asset` field. An asset hierarchy can contain up to 10,000 assets.

For example, your organization might want to renew and up-sell opportunities on products sold in the past. Similarly, your organization might want to track competitive products that exist in a customer environment that could potentially be replaced or swapped out.

Asset tracking is also useful for product support, providing detailed information to assist with product-specific support issues. For example, the `PurchaseDate` or `SerialNumber` could indicate whether a given product has certain maintenance requirements, including product recalls. Similarly, the `UsageEndDate` might indicate when the asset was removed from service or when a license or warranty expires.

If an application creates a new Asset record, it must at least specify a `Name` and either an `AccountId`, `ContactId`, or both.

SEE ALSO:

[Object Basics](#)

AssetFeed

Represents a single feed item in the feed displayed on the detail page for an asset record. This object is available in API version 18.0 and later.

An asset feed shows recent changes to an asset record for any fields that are tracked in feeds, and comments and posts about the record. It is a useful way to stay up-to-date with changes to assets.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Asset object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	Type
	textarea

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The content of the AssetFeed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type <code>int</code></p> <p>Properties <code>Filter, Group, Sort</code></p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type <code>reference</code></p> <p>Properties <code>Filter, Group, Nillable, Sort</code></p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type <code>base64</code></p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>

Field Name	Details
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if Type is ContentPost. Setting AssetFeed automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. Use FeedItem instead.</p>

Field Name	Details
InsertedById	<p>The ID of the associated FeedPost. A FeedPost represents the following types of changes in an AssetFeed: status updates, changes to tracked fields, text posts, link posts, and content posts.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code>

Field Name	Details
	<ul style="list-style-type: none"> • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p>
	 Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on the feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for the FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of the <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the asset record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion article associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as an AssetFeed object of Type ContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The title of the AssetFeed. When the Type is LinkPost, the <code>LinkUrl</code> is the URL, and this field is the link name.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of AssetFeed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence. • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field.

Field Name	Details
	<ul style="list-style-type: none"> • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. • <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case.
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • <code>Visibility</code> can be updated on record posts. • The <code>update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for an asset record.

SEE ALSO:

[Asset](#)
[Product2](#)
[FeedItem](#)

AssetOwnerSharingRule

Represents the rules for sharing an Asset with users other than the owner. This object is available in API version 33.0 and later.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AssetAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters.</p>

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Cases owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group. Target users or groups are given access.</p>

Usage

Use this object to manage the sharing rules for assets. General sharing uses this object.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

AssetShare

Represents a sharing entry on an Asset. This object is available in API version 33.0 and later.

Supported Calls

`describeSObjects(), create(), delete(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
AssetAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to the Asset. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All This value is not valid for creating or deleting records. <p>This field must be set to an access level that is higher than the organization's default access level for cases.</p>
AssetId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Asset associated with this sharing entry. This field can't be updated.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Valid values include:</p> <ul style="list-style-type: none"> • <code>Manual</code>—The User or Group has access because a user with “All” access manually shared the Asset with them. • <code>Owner</code>—The User is the owner of the Asset. • <code>Rule</code>—The User or Group has access via an Asset sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the Asset. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view and edit Asset records owned by other users.

If you attempt to create a new record that matches an existing record, request updates any modified fields and returns the existing record.

AssetTag

Associates a word or short phrase with an Asset.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

AssetTag stores the relationship between its parent TagDefinition and the Asset being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

AssignedResource

Represents a service resource who is assigned to a service appointment. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

Supported Calls

`create()`,`delete()`,`describeLayout()`,`describesObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
ActualTravelTime	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The number of minutes that the service resource needed to travel to the service appointment they're assigned to. You can enter a value with up to two decimal places.</p>
AssignedResourceNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An auto-generated number identifying the resource assignment.</p>
EstimatedTravelTime	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field Name	Details
	Description The estimated number of minutes needed for the service resource to travel to the service appointment they're assigned to. You can enter a value with up to two decimal places.
ServiceAppointmentId	Type reference Properties Create, Filter, Group, Sort Description The service appointment that the resource is assigned to.
ServiceResourceId	Type reference Properties Create, Update, Filter, Group, Sort Description The resource who is assigned to the service appointment.

Usage

You can assign multiple service resources to a service appointment. Service resources who are assigned to service appointments cannot be deactivated until they are removed from the appointments.

AssignmentRule

Represents an assignment rule associated with a Case or Lead.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `search()`

Special Access Rules

- This object is read only. Assignment rules are created, configured, and deleted in the user interface.
- Customer Portal users can't access this object.

Fields

Field	Details
Active	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this assignment rule is active (<code>true</code>) or not (<code>false</code>).</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of this assignment rule.</p>
ObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Type of assignment rule—Case or Lead.</p>

Usage

Before creating or updating a new Case or Lead, a client application can query (by name) the AssignmentRule to obtain the ID of the assignment rule to use, and then assign that ID to the `assignmentRuleId` field of the AssignmentRuleHeader. The AssignmentRuleHeader can be set using either SOAP API or REST API.

SEE ALSO:

[Object Basics](#)

AsyncApexJob

Represents an individual Apex sharing recalculation job, a batch Apex job, a method with the `future` annotation, or a job that implements `Queueable`.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ApexClassID	<p>Type reference,</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the Apex class executing the job. Label is Class ID.</p>
CompletedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The date and time when the job was completed.</p>
ExtendedStatus	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description If one or more errors occurred during the batch processing, this contains a short description of the first error. A more detailed description of that error, along with any subsequent errors, is emailed to the user who started the running batch class. This field is available in API version 19.0 and later.</p>
JobItemsProcessed	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of job items processed. Label is Batches Processed.</p>
JobType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of job being processed. Valid values are:</p> <ul style="list-style-type: none"> • Future • SharingRecalculation

Field Name	Details
	<ul style="list-style-type: none"> ScheduledApex BatchApex BatchApexWorker TestRequest TestWorker ApexToken Queueable
MethodName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the Apex method being executed. Label is <code>Apex Method</code>.</p>
NumberOfErrors	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total number of batches with a failure. A batch is considered transactional, so any unhandled exceptions constitute an entire failure of the batch. Label is <code>Failures</code>.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The status of the job. Valid values are:</p> <ul style="list-style-type: none"> Holding¹ Queued Preparing Processing Aborted Completed Failed
TotalJobItems	<p>Type int</p>

¹ This status applies to batch jobs in the Apex flex queue.

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description Total number of batches processed. Each batch contains a set of records. Label is Total Batches.</p>

Usage

Use this object to query Apex batch jobs in your organization.

AttachedContentDocument

This read-only object contains all `ContentDocument` objects associated with an object.

Supported Calls

```
describeSObjects()
```

Fields

Field Name	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the attached <code>ContentDocument</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the document in bytes.</p>
ContentUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	URL for links and Google Docs. This field is set only for links and Google Docs, and is one of the fields that determine the <code>FileType</code> .
	This field is available in API version 31.0 and later.
ExternalDataSourceName	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Name of the external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.
	This field is available in API version 32.0 and later.
ExternalDataSourceType	Type
	picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	Type of external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.
	This field is available in API version 35.0 and later.
FileExtension	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	File extension of the attached <code>ContentDocument</code> .
	This field is available in API version 31.0 and later.
FileType	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Type of document, determined by the file extension.
LinkedEntityId	Type
	reference

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the record the ContentDocument is attached to.</p>
SharingOption	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Controls whether or not sharing is frozen for a file. Only administrators and file owners with Collaborator access to the file can modify this field. Default is Allowed, which means that new shares are allowed. When set to Restricted, new shares are prevented without affecting existing shares. This field is available in API versions 35.0 and later.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Title of the attached ContentDocument.</p>

Usage

Use this object to list all ContentDocument objects attached to an object via a feed post.

To retrieve ContentDocument objects, issue a describe call on an object, which returns a query result for each activity since the record was created. You can't directly query this object.

AttachedContentNote

This read-only object contains all ContentNote objects associated with an object. This object is available in API version 35.0 and later.

Supported Calls

`describeSObjects()`

Special Access Rules

- Notes must be enabled.

- Chatter must be enabled.

Fields

Field Name	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the attached ContentNote</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the note in bytes.</p>
FileExtension	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description File extension of the attached ContentNote.</p>
FileType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Type of file for the note. All notes have a file type of SNOTE.</p>
LinkedEntityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the record the ContentNote is attached to.</p>

Field Name	Details
TextPreview	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A preview of the note, up to 255 characters.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Title of the note.</p>

Usage

Use this object to list all `ContentNote` objects attached to an object.

To retrieve `ContentNote` objects, issue a describe call on an object, which returns a query result for each note created or attached. You can't directly query this object.

Attachment

Represents a file that a User has uploaded and attached to a parent object.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Body	<p>Type base64</p> <p>Properties Create, Update</p> <p>Description Required. Encoded file data.</p>

Field	Details
BodyLength	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the file (in bytes).</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
ContentType	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The content type of the attachment. If the <code>Don't allow HTML uploads as attachments or document</code> security setting is enabled for your organization, you cannot upload files with the following file extensions: <code>.htm</code>, <code>.html</code>, <code>.htt</code>, <code>.htx</code>, <code>.mhtm</code>, <code>.mhtml</code>, <code>.shtm</code>, <code>.shtml</code>, <code>.acgi</code>, <code>.svg</code>.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description Description of the attachment. This field is available in API version 18.0 and later.</p>
IsEncrypted	<p> Note: This information is about Shield Platform Encryption and not Classic Encryption.</p> <p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the attachment is encrypted using Shield Platform Encryption (<code>true</code>) or not (<code>false</code>). This field is available in API version 34.0 and later.</p>
IsPartnerShared	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Update</p> <p>Description Indicates whether this record is shared with a connection using Salesforce to Salesforce. Label is <code>Is Shared With Partner</code>.</p>
IsPrivate	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this record is viewable only by the owner and administrators (<code>true</code>) or viewable by all otherwise-allowed users (<code>false</code>). During a create or update call, it is possible to mark an Attachment record as private even if you are not the owner. This can result in a situation in which you can no longer access the record that you just inserted or updated. Label is Private. Attachments on tasks or events can't be marked private.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of the attached file. Label is File Name.</p>
OwnerId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the User who owns the attachment. This field was required previous to release 9.0. Beginning with release 9.0, it can be null on create. The owner of an attachment on a task or event must be the same as the owner of the task or event.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the parent object of the attachment. The following objects are supported as parents of attachments:</p> <ul style="list-style-type: none">• Account• Asset• Campaign• Case• Contact• Contract• Custom objects• EmailMessage• EmailTemplate• Event• Lead• Opportunity• Product2• Solution• Task



Note: If you are importing Attachment data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. For example, for compliance reasons, you may prefer to set the `CreatedDate` to the date the record was originally created in your system, rather than the date it was imported into Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself.

Usage

The API sends and receives the binary file attachment data encoded as a base64Binary data type. Prior to creating a record, client applications must encode the binary attachment data as base64. Upon receiving a response, client applications must decode the base64 data to binary (this conversion is usually handled for you by the SOAP client).

The create call restricts these files to a maximum size of 25 MB. For a file attached to a Solution, the limit is 1.5 MB. The maximum email attachment size is 3 MB.

The API supports attachments on email in create, delete, or update calls. The query call does not return attachments parented by email, unless the user performing the query has the "Modify All Data" permission.



Note:

- Attachment records are not searched during text searches.
- When issued by an administrator, the query results include Attachment records from the Recycle Bin.
- When issued by a non-administrator, the `queryAll()` call results do not include Attachment records from the Recycle Bin.

Access to fields depends on the method being used:

- All of the fields are accessible using the `describeSObjects()` and `query()` calls. With the `create()` call, you can insert the `Name`, `ParentId`, `Body`, `IsPrivate`, and `OwnerId` fields.
- To modify existing records, the `update()` call gives you access to change the `Name`, `Body`, `IsPrivate`, and `OwnerId` fields.
- You can access all of the fields using a `query()` call. However, you can't receive the `Body` field for multiple records in a single `query()` call. If your query returns the `Body` field, your client application must ensure that only one row with one Attachment is returned; otherwise, an error occurs. A more effective approach is to return IDs (but not Attachment records in the `Body` field) from a `query()` call and then pass them into `retrieve()` calls that return the `Body` field.
- For information about accessing the attachments of archived activities, see Archived Activities.

SEE ALSO:

[Note](#)

AuraDefinition

Represents a Lightning definition, such as component markup, a client-side controller, or an event. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
<code>AuraDefinitionBundleId</code>	Type reference

Field Name	Details
	<p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the bundle containing the definition. A bundle contains a Lightning definition and all its related resources.</p>
DefType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The definition type. Valid values are:</p> <ul style="list-style-type: none"> • APPLICATION — Lightning Components app • CONTROLLER — client-side controller • COMPONENT — component markup • EVENT — event definition • HELPER — client-side helper • INTERFACE — interface definition • RENDERER — client-side renderer • STYLE — style (CSS) resource • PROVIDER — reserved for future use • MODEL — deprecated, do not use • TESTSUITE — reserved for future use • DOCUMENTATION — documentation markup • TOKENS — tokens collection • DESIGN — design definition • SVG — SVG graphic resource
Format	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The format of the definition. Valid values are:</p> <ul style="list-style-type: none"> • XML for component markup • JS for JavaScript code • CSS for styles

Field Name	Details
Source	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The contents of the Lightning definition. This is all the markup or code for the definition.</p>

Usage

For more information, see the [Lightning Components Developer Guide](#).

AuraDefinitionBundle

Represents a Lightning definition bundle, such as a component or application bundle. A bundle contains a Lightning definition and all its related resources. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ApiVersion	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The API version for this bundle. Every bundle has an API version specified at creation.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text description of the bundle. Maximum size of 255 characters.</p>

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the record in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This field is automatically generated but you can supply your own value if you create the record using the API.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the MasterLabel.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Master label for the Lightning bundle. This internal label doesn't get translated.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This

Field Name	Details
	<p>field's value is the namespace prefix of the Developer Edition organization of the package developer.</p> <ul style="list-style-type: none"> In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

For more information, see the [Lightning Components Developer Guide](#).

AuthConfig

Represents authentication options for a community or custom domain that was created by using My Domain. This object is available in API version 32.0 and later.

The fields for this object control the login options that show up on the login page for users of a community or custom domain.

- Logging in with a username and password
- Using SAML for single sign-on
- Authentication provider logins from a third-party service, such as Facebook or Twitter

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

You must have "View Setup and Configuration" permission to view the settings.

Fields

Field Name	Details
<code>AuthOptionsAuthProvider</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description If <code>true</code>, at least one Auth. Provider is selected to show up on the login page, and this object has child <code>AuthConfigProvider</code> objects for each provider.</p>
<code>AuthOptionsSaml</code>	<p>Type boolean</p>

Field Name	Details
	Properties Filter
	Description
	<p>If <code>true</code>, at least one SAML configuration is selected to show up on the login page. If the organization has only one SAML configuration, this value indicates whether that configuration is selected to show up on the login page. If the organization has multiple SAML configurations, see the child <code>AuthConfigProvider</code> objects for each configuration.</p>
AuthOptionsUsernamePassword	Type <code>boolean</code>
	Properties Filter
	Description
	<p>If <code>true</code>, the login option for a username and password appears on the login page.</p>
DeveloperName	Type <code>string</code>
	Properties Filter , Group , Sort
	Description
	<p>The name of the domain created using My Domain or, for communities, a concatenated string of <code>community_name_community_prefix</code>.</p>
IsActive	Type <code>boolean</code>
	Properties Defaulted on create , Filter , Group , Sort
	Description
	<p>Whether this configuration is in use.</p>
Language	Type <code>picklist</code>
	Properties Filter , Group , Restricted picklist , Sort
	Description
	<p>The language for the organization.</p>
MasterLabel	Type <code>string</code>

Field Name	Details
NamespacePrefix	<p>Properties Filter, Group, Sort</p> <p>Description The text that's used to identify the Visualforce page in Setup.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Type	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The organization type for this object.</p> <ul style="list-style-type: none"> • Org (includes custom domains) • Community • Site • Portal
Url	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p>

Field Name	Details
Description	
The login URL of the organization for this AuthConfig object. Each URL has only one associated AuthConfig object.	

AuthConfigProviders

Represents an authentication provider that's configured in an organization. This object is a child of the AuthConfig object. This object is available in API version 32.0 and later.

This object links the authentication configuration for an organization to the Auth. Provider through the `AuthOptionsAuthProvider` field of the AuthConfig object. The login page of a Community or custom domain that was created by using My Domain can allow multiple SAML configurations and multiple authentication providers. These configurations can be set to show up as buttons on the login page. Each configuration has an AuthConfigProvider object.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

You must have "View Setup and Configuration" permission to view the settings.

Fields

Field Name	Details
AuthConfigId	Type reference Properties Filter, Group, Sort Description The ID for this configuration.
AuthProviderId	Type reference Properties Filter, Group, Sort Description The ID of the Auth. Provider or SAML configuration.

AuthProvider

Represents an authentication provider (or auth provider) in your organization.

An auth provider enables users to log into your Salesforce organization using their login credentials from an external service provider such as Facebook[®] or Janrain[®].

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Only users with the “Customize Application” and “Manage AuthProviders” permissions can access this object.

Fields

Field Name	Details
AuthorizeUrl	<p>Type url</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Required, but only if <code>ProviderType</code> is <code>OpenIdConnect</code>. The OAuth authorization endpoint URL. Used only with OpenID Connect authentication providers. Available in API version 29.0 and later.</p>
ConsumerKey	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Required. The app’s key that is registered at the third-party Single Sign-On provider.</p>
ConsumerSecret	<p>Type string</p> <p>Properties Create, Nillable</p> <p>Description Required. The consumer secret of the app that is registered at the third-party Single Sign-On provider. This field cannot be updated. When using <code>create()</code> this field must be encrypted. To create an encrypted form of the consumer secret from plain text:</p>

Field Name	Details
	<p>1. Create an auth provider with the <code>ConsumerSecret</code> plain text value.</p> <p>2. Save the auth provider.</p> <p>3. Create an outbound change set that includes the auth provider component.</p> <p>The new change set .xml file will have an entry in the form <code><consumerSecret>++XYZ++</consumerSecret></code> where ++XYZ++ is the encrypted secret.</p>
CustomMetadataTypeRecord	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Required, but only with custom authentication provider plug-ins, when <code>ProviderType</code> is <code>Custom</code>. The API name of the authentication provider. Available in API version 36.0 and later.</p>
DefaultScopes	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Required, but only if <code>ProviderType</code> is <code>OpenIdConnect</code>. The scopes to be sent with the authorization request, if not specified when a flow is started. Used only with OpenID Connect authentication providers. Available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Used when referring to the auth provider from a program.</p>
ErrorUrl	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description A custom error URL for the provider to use to report any errors.</p>

Field Name	Details
ExecutionUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The user that runs the Apex handler class. The user must have the “Manage Users” permission. A user is required if you specify a registration handler class.</p>
FriendlyName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. A user-friendly name for the provider.</p>
IconUrl	<p>Type url</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description The path to an icon to use as a button on the login page. Users click the button to log in with the associated auth provider.</p>
IdTokenIssuer	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Only available if ProviderType is OpenIdConnect. This value identifies the source of the authentication token in the form <i>https://URI</i>. Used only with OpenID Connect authentication providers. If provided, Salesforce validates the returned <code>id_token</code> value. The OpenID Connect specification requires an <code>id_token</code> value to be returned with the <code>access_token</code> value. Available in API version 30.0 and later.</p>
LogoutUrl	<p>Type url</p> <p>Properties Create, Filter, Nullable, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>Provides a specific destination for users after they log out if they authenticated using social sign-on. The URL must be fully qualified with an <code>http</code> or <code>https</code> prefix, such as <code>https://acme.my.salesforce.com</code>. Available in API version 33.0 and later.</p>
OptionsIncludeOrgIdInId	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Includes the organization ID to differentiate between users with the same user ID from two sources (such as two sandboxes). Only available for Salesforce authentication providers. Once set to <code>true</code>, it can't be set to <code>false</code>. Available in API version 32.0 and later.</p>
OptionsSendAccessTokenInHeader	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Required only if <code>ProviderType</code> is <code>OpenIdConnect</code>. When <code>true</code>, the access token is sent to the <code>UserinfoUrl</code> in a header instead of a query string. Used only with OpenID Connect authentication providers. Available in API version 30.0 and later.</p>
OptionsSendClientCredentialsInHeader	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Required only if <code>ProviderType</code> is <code>OpenIdConnect</code>. When <code>true</code>, the client credentials are sent in a header, instead of a query string, to the <code>tokenUrl</code>. The credentials are in the standard OpenID Connect Basic Credentials header form, which is <code>Basic <token></code>, where <code><token></code> is the base64-encoded string <code>"clientkey:clientsecret"</code>. Used only with OpenID Connect authentication providers. Available in API version 30.0 and later.</p>
ProviderType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	Description
	<p>Required. The third-party Single Sign-On provider to use. Valid values are:</p> <ul style="list-style-type: none"> • Facebook • Salesforce • Janrain • LinkedIn (Available in API version 32.0 and later.) • Twitter (Available in API version 32.0 and later.) • OpenIdConnect (Available in API version 29.0 and later.)
	 Note: This type requires values for the following fields:
	<ul style="list-style-type: none"> — AuthorizeUrl — DefaultScopes — TokenUrl — UserInfoUrl
	<ul style="list-style-type: none"> • MicrosoftACS Microsoft Access Control Service typically provides authentication for a Microsoft Office 365 service like SharePoint® Online. (Available in API version 31.0 and later.) • GitHub—Use the GitHub provider to log in users of your Force.com app to GitHub using OAuth. When logged in to GitHub, your app can make calls to GitHub APIs. The GitHub provider isn't available as a single sign-on provider, which means users can't log in to your Salesforce organization using their GitHub login credentials. (Available in API version 35.0 and later.) • Custom (Available in API version 36.0 and later.)
PluginId	Type reference
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description An existing Apex class that implements the <code>Auth.AuthProviderPlugin</code> interface. Available in API version 36.0 and later.
RegistrationHandlerId	Type reference
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description An existing Apex class that implements the <code>Auth.RegistrationHandler</code> interface.

Field Name	Details
TokenUrl	<p>Type url</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Required, but only if ProviderType is OpenIdConnect. The OAuth token endpoint URL. Used only with OpenID Connect authentication providers. Available in API version 29.0 and later.</p>
UserInfoUrl	<p>Type url</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Required, but only if Provider-type is OpenIdConnect. The OpenID Connect endpoint URL. Used only with OpenID Connect authentication providers. Available in API version 29.0 and later.</p>

AuthSession

The AuthSession object represents an individual user session in your organization. This object is available in versions 29.0 and later.

Supported Calls

`delete(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description The date and time this session was created. This field is a standard system field.</p>
IsCurrent	<p>Type boolean</p>

Field Name	Details
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	If true, the session is a member of the user's current session family. This field is available in API version 37.0 and later.
LastModifiedDate	Type
	dateTime
	Properties
	Defaulted on create, Filter, Sort
	Description
	The date and time this session was last updated. A session expires when the current date and time equals LastModifiedDate + NumSecondsValid. This field is a standard system field.
LoginGeoId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	The 18-character ID for the record of the geographic location of the user for a login event. Due to the nature of geolocation technology, the accuracy of geolocation fields (for example, country, city, postal code) can vary. This field is available in API version 34.0 and later.
LoginHistoryId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	The 18-character ID for a successful login event. When a session is reused, Salesforce updates the LoginHistoryId with the value from the most recent login. This field is available in API version 33.0 and later.
LoginType	Type
	picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	The type of login, for example, Application.

Field Name	Details
LogoutUrl	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The page or view to display after users log out of a community, or an organization if they authenticated using SAML. This field is available in API version 32.0 and later.</p>
NumSecondsValid	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of seconds before the session expires, starting from the last update time.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The 18-character ID for the parent session, if one exists (for example, if the current session is for a canvas app). If the current session doesn't have a parent, this value is the current session's own ID.</p>
SessionSecurityLevel	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Standard or High, depending upon the authentication method used.</p>
SessionType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of session. Common ones are UI, Content, API, and Visualforce.</p>
SourceIp	<p>Type string</p>

Field Name	Details
	Properties
	Filter, Group, Sort
	Description
	IP address of the end user's device from which the session started. This address can be an IPv4 or IPv6 address.
UserType	Type
	picklist
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	The kind of user for this session. Types include Standard, Partner, Customer Portal Manager, High Volume Portal, and CSN Only.
UserId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	The user's Salesforce user ID.

Usage

The AuthSession object exposes session data and enables read and delete operations on that data. For example, use this object to create a report showing who is signed in to your organization, or to create a tool to delete a session, ending that user's session. For a user, only their own sessions are available, while administrators can see all sessions.

You can't change user sessions with this object. You can only read and delete them.

BackgroundOperation

Represents a background operation in an asynchronous job queue. This object is available in API version 35.0 and later.

Supported Calls

```
describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()
```

Fields

Field Name	Details
Error	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The error message for the operation. Applies only if the operation has an error status.</p>
ExecutionGroup	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Applies only if the operation is merged with other operations into an execution group to be processed in bulk. Identifies the execution group.</p>
ExpiresAt	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description After this time, the operation is removed from the asynchronous job queue. Applies only if the operation has a status of complete, canceled, error, or merged.</p>
FinishedAt	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description When the operation reached the status of completed or error.</p>
GroupLeaderId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Applies only if the operation is merged with other operations into an execution group to be processed in bulk. Identifies the operation that's selected as the leader of the execution group.</p>

Field Name	Details
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description Identifies the background operation.</p>
NumFollowers	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Applies only if the operation is merged with other operations into an execution group to be processed in bulk. Number of other operations that are in the execution group.</p>
ParentKey	<p>Type string</p> <p>Properties Filter, Group, idLookup, Nillable, Sort</p> <p>Description Tag that identifies related sets of operations, if any.</p>
ProcessAfter	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The operation is scheduled to be processed after this time.</p>
RetryBackoff	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Applies only if the operation has an error status. The first retry is attempted immediately. Each subsequent retry is increasingly delayed according to an exponential expression that's multiplied by the <code>RetryBackoff</code>, in milliseconds. Specifically, the delay time is $(2^n - 1) \times R$, where n is the <code>RetryCount</code>, and R is the <code>RetryBackoff</code>. The default value for <code>RetryBackoff</code> depends on the type of operation. For example, the <code>RetryBackoff</code> default for write operations on external objects</p>

Field Name	Details
RetryCount	<p>is 1,000 milliseconds. For write operations, retries are attempted immediately, after 3 seconds, after 7 seconds, after 15 seconds, and so on.</p>
RetryLimit	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number of attempted retries. Applies only if the operation has an error status.</p>
SequenceGroup	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Identifies the sequence group. Applies only if the operation is merged with other operations into an execution group to be processed in bulk. Within an execution group, operations can be placed into a sequence group to be executed in a specific order.</p>
SequenceNumber	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Order position within the sequence group. Applies only if the operation is merged with other operations into an execution group to be processed in bulk. Within an execution group, operations can be placed into a sequence group to be executed in a specific order.</p>
StartedAt	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	Description
	When the operation started running.
Status	Type picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	Status of the background operation. The options are:
	<ul style="list-style-type: none"> ● New ● Scheduled ● Canceled ● Merged ● Waiting ● Running ● Error ● Complete
SubmittedAt	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	When the operation was added to the job queue.
Timeout	Type int
	Properties
	Filter, Group, Nillable, Sort
	Description
	Maximum time in milliseconds to wait for results after the operation started running.
WorkerUri	Type string
	Properties
	Filter, Group, idLookup, Nillable, Sort
	Description
	URI of the worker that performed the operation.

Field Name	Details
	<p>Example for a Salesforce Connect OData operation:</p> <pre>services/data/v35.0/xds/upsert</pre>

Usage

The `BackgroundOperation` object lets you:

- Monitor the job status of asynchronous operations.
- View errors that are related to the asynchronous operations.
- Extract statistics for the asynchronous job queue.

BackgroundOperationResult

Stores error messages generated when running Async SOQL queries or importing data into BigObjects using Bulk API. This is a `BigObject`, available in API version 37.0 and later.

Each instance of `BackgroundOperationResult` represents one error. The `Message` field stores the text of the error message. The `ParentID` field stores the:

- job ID of the query, in case of Async SOQL
- batch ID for the data import, in case of Bulk API

Bulk API validates data at the time of import, and generates an error message for the first occurrence of invalid data in any row of the data file. The validation performed depends on the type of data being imported.

- **Text**—The length of the input string must be less than or equal to the length of the corresponding text field in the target object.
- **Number**—The input data must be a number, whose scale and precision are compatible with the corresponding number field in the target object.
- **ID**—The input data must be a valid 15- or 18-character ID.
- **DateTime**—The input data must be a valid `dateTime` value, in the approved format.
- **Lookup**—The lookup value must be a valid 15- or 18-character ID.

Supported Calls

```
describeSObjects(), query()
```

Fields

Field Name	Details
<code>CreatedBy</code>	<p>Type ID</p> <p>Properties Nillable</p>

Field Name	Details
	<p>Description The user ID of the user initiating the Bulk API or Async SOQL request.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create</p> <p>Description The date and time at which the Bulk API or Async SOQL request was made.</p>
Data	<p>Type string</p> <p>Properties Nillable</p> <p>Description The data that generated the error message. The total length is limited to 2,000 characters, and each column can occupy a maximum of 50 characters. Any data exceeding those limits is truncated.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, idLookup</p> <p>Description The ID of the error message.</p>
Message	<p>Type string</p> <p>Properties Nillable</p> <p>Description The text of the error message.</p>
MessageType	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The type of error message. The possible values are: ERROR, WARNING, or INFO.</p>
ParentId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Sort</p>
	<p>Description The batch ID, in case of Bulk API, or the job ID, in case of Async SOQL.</p>

Usage

You can check for errors by querying the `BackgroundOperationResult` object. For example, this query returns details of all errors in a data file imported using Bulk API, whose batch ID is `751xx000000006OAAQ`.

```
SELECT CreatedById, CreatedDate, Id, Message, MessageType, ParentId FROM
BackgroundOperationResult WHERE ParentId = "751xx000000006OAAQ"
```

 **Note:** You can only view errors resulting from Async SOQL or Bulk API requests that you initiated, unless you have the global permission to view all data.

Bookmark

Represents a link between opportunities that share common information.

This object is available to organizations with the Similar Opportunities feature enabled.

Supported Calls

```
create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()
```

Fields

Field	Details
ID	<p>Type ID</p> <p>Properties Defaulted on create, Filter</p> <p>Description ID of the bookmark. Label is Bookmark ID.</p>
FromId	<p>Type ID</p> <p>Properties Filter</p> <p>Description The originating opportunity. Label is Bookmarked From ID</p>

Field	Details
ToId	<p>Type ID</p> <p>Properties Filter</p> <p>Description The opportunity to which the originating opportunity is linked. Label is Bookmarked To ID.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>

Usage

The Bookmark object works with the Opportunity object only.

Use this read-only object to query the bookmarks between opportunities in your organization. In the online application, users can search for opportunities that share attributes with their opportunity. The user can then bookmark the appropriate opportunities for future reference.

BrandTemplate

Letterhead for HTML EmailTemplate.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the letterhead. Limited to 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Letterhead Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the letterhead is available for use (<code>true</code>) or not (<code>false</code>). Label is Active.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the template as it appears in the user interface. Limited to 255 characters. Label is Brand Template Name.</p>
NamespacePrefix	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>
Value	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The contents of the letterhead, in HTML, including any logos.</p>

Usage

Use this object to brand EmailTemplate records with your letterhead. You can also set a brand template to active or inactive. For example, if you have five different marketing brands, you can maintain each different brand in one template, and assign to the appropriate EmailTemplate.

SEE ALSO:

[EmailTemplate](#)

BusinessHours

Specifies the business hours of your support organization. Escalation rules are run only during these hours.

If business hours are associated with any Holiday records, then business hours and escalation rules associated with business hours are suspended during the dates and times specified as holidays.

Supported Calls

`create(), getUpdated(), query(), retrieve(), update()`

Special Access Rules

Customer Portal users can't access this object.

All users, even those without the "View Setup and Configuration" user permission, can view business hours via the API.

Fields

Field	Details
BusinessHoursId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the BusinessHours associated with the SlaProcess.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the business hours is active (<code>true</code>) or not active (<code>false</code>).</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the business hours.</p>
IsDefault	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the business hours are set as the default business hours (<code>true</code>) or not (<code>false</code>).</p>
FridayEndTime	<p>Type time</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
FridayStartTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business opens.</p>
MondayEndTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
MondayStartTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business opens.</p>
SaturdayEndTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
SaturdayStartTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business opens.</p>
SundayEndTime	<p>Type time</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
SundayStartTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business opens.</p>
ThursdayEndTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
ThursdayStartTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business opens.</p>
TimeZoneSidKey	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
	<p>Description The time zone of the business hours.</p>
TuesdayEndTime	<p>Type time</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Time that business closes.</p>
TuesdayStartTime	<p>Type time</p>

Field	Details
	Properties Create, Filter, Nillable, Sort, Update Description Time that business opens.
WednesdayEndTime	Type time Properties Create, Filter, Nillable, Sort, Update Description Time that business closes.
WednesdayStartTime	Type time Properties Create, Filter, Nillable, Sort, Update Description Time that business opens.

Usage

Use this object to specify the business hours at which your support team operates. Escalation rules only run during the business hours with which they are associated. To set business hours to 24-hours a day, set the times from midnight to midnight (00:00:00 ~ 00:00:00) on each day.

By default, business hours are set from 12:00 AM to 12:00 AM in the default time zone specified in your organization's profile.

SEE ALSO:

[Object Basics](#)

BusinessProcess

Represents a business process.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users cannot access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Description of this business process. Limit: 255 characters.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this business process can be presented to users in the Salesforce user interface (<code>true</code>) or not (<code>false</code>) when creating a new record type or changing the business process of an existing record type.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of this business process. Limit: 80 characters.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.

Field	Details
	<ul style="list-style-type: none"> In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
TableEnumOrId	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Required. One of the following values: Case, Opportunity, or Solution. Label is Entity Enumeration Or ID.</p>

Usage

Use the BusinessProcess object to offer different subsets of picklist values to different users for the LeadStatus, CaseStatus, and OpportunityStage fields. Similar to a RecordType, a BusinessProcess identifies the type of a row in a Case, Lead, or Opportunity and implies a subset of picklist values for these three fields. The values for the remaining picklist fields are driven off of RecordType.

SEE ALSO:

[Object Basics](#)

CallCenter

Represents a call center, which is a logical representation of a single computer-telephony integration (CTI) system instance in an organization.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AdapterURL	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort</p>

Field	Details
	Description
	<p>An optional field that specifies the location of where the CTI adapter is hosted. For example, <code>http://localhost:11000</code>.</p>
	<p>This field is available for call centers using CTI Toolkit version 4.0 and API version 23.0 or later.</p>
CustomSettings	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort
	Description
	<p>Specifies settings in the call center definition file, such as whether the call center uses the Open CTI, and SoftPhone properties, such as height in pixels.</p>
	<p>This field is available for Open CTI and in API version 25.0 or later.</p>
Id	Type
	ID
	Properties
	Defaulted on create, Filter
	Description
	<p>System field that uniquely identifies this call center. Label is Call Center ID. This ID is created automatically when the call center is created.</p>
InternalName	Type
	string
	Properties
	Create, Filter, Group, Sort
	Description
	<p>The internal name of the call center.</p>
	<p>Limit is 80 characters.</p>
Name	Type
	string
	Properties
	Create, Filter, Group, Sort
	Description
	<p>The name of the call center.</p>
	<p>Limit is 80 characters.</p>
Version	Type
	double

Field	Details
	<p>Properties Create, Filter, Nillable, Sort</p>
	<p>Description The version of the CTI Developer's Toolkit used to create the call center (for versions 2.0 and later). This field is available in API version 18.0 and later.</p>

Usage

Create a call center or query an existing call center.

Campaign

Represents and tracks a marketing campaign, such as a direct mail promotion, webinar, or trade show.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ActualCost	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Amount of money spent to run the campaign.</p>
AmountAllOpportunities	<p>Type currency</p> <p>Properties Filter, Sort</p>

Field	Details
	<p>Description Amount of money in all opportunities associated with the campaign, including closed/won opportunities. Label is Total Value Opportunities.</p>
AmountWonOpportunities	<p>Type currency</p> <p>Properties Filter, Sort</p> <p>Description Amount of money in closed or won opportunities associated with the campaign. Label is Total Value Won Opportunities.</p>
BudgetedCost	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Amount of money budgeted for the campaign.</p>
CampaignMemberRecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record type ID for CampaignMember records associated with the campaign.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Restricted picklist, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the campaign. Limit: 32 KB. Only the first 255 characters display in reports.</p>

Field	Details
EndDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Ending date for the campaign. Responses received after this date are still counted.</p>
ExpectedResponse	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Percentage of responses you expect to receive for the campaign.</p>
ExpectedRevenue	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Amount of money you expect to generate from the campaign.</p>
HierarchyActualCost	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Calculated field for the total amount of money spent to run the campaigns in a campaign hierarchy. Label is Total Actual Cost in Hierarchy.</p>
HierarchyBudgetedCost	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Calculated field for the total amount of money budgeted for the campaigns in a campaign hierarchy. Label is Total Budgeted Cost in Hierarchy.</p>
HierarchyExpectedRevenue	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description Calculated field for the total amount of money you expect to generate from the campaigns in a campaign hierarchy. Label is Total Expected Revenue in Hierarchy.</p>
HierarchyNumberSent	<p>Type int</p> <p>Properties Filter</p> <p>Description Calculated field for the total number of individuals targeted by the campaigns in a campaign hierarchy. For example, the number of email messages sent. Label is Total Num Sent in Hierarchy.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this campaign is active (<code>true</code>) or not (<code>false</code>). Default value is <code>false</code>. Label is Active.</p>
LastActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Value is one of the following, whichever is the most recent: <ul style="list-style-type: none"> • Due date of the most recent event logged against the record. • Due date of the most recently closed task associated with the record. </p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of the campaign. Limit: is 80 characters.</p>
NumberOfContacts	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of contacts associated with the campaign. Label is Total Contacts.</p>
NumberOfConvertedLeads	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of leads that were converted to an account and contact due to the marketing efforts in the campaign. Label is Converted Leads.</p>
NumberOfLeads	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of leads associated with the campaign. Label is Total Leads.</p>
NumberOfOpportunities	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of opportunities associated with the campaign. Label is Num Total Opportunities.</p>
NumberOfResponses	<p>Type int</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description Number of contacts and unconverted leads with a Member Status equivalent to "Responded" for the campaign. Label is Total Responses.</p>
NumberOfWonOpportunities	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of closed or won opportunities associated with the campaign. Label is Num Won Opportunities.</p>
NumberSent	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Number of individuals targeted by the campaign. For example, the number of emails sent. Label is Num Sent.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns this campaign. Default value is the user logging in to the API to perform the create.</p>
ParentCampaign	<p>Type reference</p> <p>Properties Create, Filter Nillable, Update</p> <p>Description The campaign above the selected campaign in the campaign hierarchy.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description ID of the parent Campaign record, if any.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Starting date for the campaign.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Status of the campaign, for example, Planned, In Progress. Limit: 40 characters.</p>
TotalAmountAllOpportunities	<p>Type currency</p> <p>Properties Filter</p> <p>Description Calculated field for total amount of all opportunities associated with the campaign hierarchy, including closed/won opportunities. Label is Total Value Opportunities in Hierarchy.</p>
TotalAmountAllWonOpportunities	<p>Type currency</p> <p>Properties Filter</p> <p>Description Calculated field for amount of all closed/won opportunities associated with the campaign hierarchy. Label is Total Value Won Opportunities in Hierarchy.</p>
TotalNumberofContacts	<p>Type int</p>

Field	Details
	<p>Properties Filter</p> <p>Description Calculated field for number of contacts associated with the campaign hierarchy. Label is Total Contacts in Hierarchy.</p>
TotalNumberOfConvertedLeads	<p>Type int</p> <p>Properties Filter</p> <p>Description Calculated field for the total number of leads associated with the campaign hierarchy that were converted into accounts, contacts, and opportunities. Label is Total Converted Leads in Hierarchy.</p>
TotalNumberOfLeads	<p>Type int</p> <p>Properties Filter</p> <p>Description Calculated field for total number of leads associated with the campaign hierarchy. This number also includes converted leads. Label is Total Leads in Hierarchy.</p>
TotalNumberOfOpportunities	<p>Type int</p> <p>Properties Filter</p> <p>Description Calculated field for the total number of opportunities associated with the campaign hierarchy. Label is Total Opportunities in Hierarchy.</p>
TotalNumberOfResponses	<p>Type int</p> <p>Properties Filter</p> <p>Description Calculated field for number of contacts and unconverted leads that have a Member Status equivalent to "Responded" for the campaign hierarchy. Label is Total Responses in Hierarchy.</p>
TotalNumberOfWonOpportunities	<p>Type int</p>

Field	Details
	<p>Properties Filter</p> <p>Description Calculated field for the total number of won opportunities associated with the campaign hierarchy. Label is Total Won Opportunities in Hierarchy.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Type of campaign, for example, Direct Mail or Referral Program. Limit: 40 characters.</p>

Usage

Client applications can create, update, delete, and query Attachment records associated with a campaign via the API.

The Campaign object is defined only for those organizations that have the marketing feature enabled and valid marketing licenses. In addition, it is accessible only to those users that are enabled as marketing users. If the organization does not have the marketing feature or valid marketing licenses, this object does not appear in the `describeGlobal()` call, and you can't use `describeSObjects()` or `query()` with the Campaign object.

 **Note:** The main constituent of a campaign is a CampaignMember. You will commonly need to update campaigns with CampaignMember.

SEE ALSO:

[Object Basics](#)

CampaignFeed

Represents a single feed item in the feed on a campaign record detail page. This object is available in API version 18.0 and later.

A campaign feed shows recent changes to a campaign record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to campaigns.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”

- “Modify All” on the Campaign object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of CampaignFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with “CanApproveFeedPostAndComment” or “ModifyAllData” permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator and the ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>

Field	Details
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>

Field	Details
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p>

Field	Details
	<p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>

Field	Details
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the campaign record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>

Field	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL. PollPost—a poll posted on a feed. ProfileSkillPost—generated when a skill is added to a user's Chatter profile. QuestionPost—generated when a user posts a question. ReplyPost—generated when Chatter Answers posts a reply. RipplePost—generated when a user creates a Thanks badge in Work.com.

Field	Details
	<ul style="list-style-type: none">• <code>TextPost</code>—a direct text entry on a feed.• <code>TrackedChange</code>—a change or group of changes to a tracked field.• <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none">• <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a campaign record.

SEE ALSO:

[Campaign](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

CampaignInfluence

Represents the association between a campaign and an opportunity in Customizable Campaign Influence. This object is available in API version 37.0 and later.

 **Note:** This information applies only to Customizable Campaign Influence and not to the original version of Campaign Influence.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

To access this object, Customizable Campaign Influence must be enabled. Customer Portal users can't access this object.

Fields

Field Name	Details
CampaignId	Type reference Properties Create, Filter, Group, Sort Description The ID of the associated campaign.
ContactId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The ID of the contact on the associated opportunity.

Field Name	Details
Influence	<p>Type percent</p> <p>Properties Create, Defaulted on create, Filter, Nullable, Sort, Update</p> <p>Description The percentage of the associated opportunity's Amount field attributed to the associated campaign.</p>
ModelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the campaign influence model associated with the record.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the associated opportunity.</p>
RevenueShare	<p>Type currency</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The amount of revenue from the associated opportunity attributed to the associated campaign.</p>

CampaignInfluenceModel

This read-only object represents a campaign influence model in Customizable Campaign Influence. Use campaign influence models to group `CampaignInfluence` records created by a specific set of triggers and workflows that you define. The Salesforce influence model is the default model. This object is available in API version 37.0 and later.



Note: This information applies only to Customizable Campaign Influence and not to the original version of Campaign Influence.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

To access this object, Customizable Campaign Influence must be enabled. Customer Portal users can't access this object.

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The API name of the influence model. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
IsDefaultModel	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the model is the primary model (<code>true</code>) or not (<code>false</code>). CampaignInfluence records associated with the primary model appear in the Campaign Influence related list on opportunities, the Influenced Opportunities related list on campaigns, and the Campaign Statistics section on campaigns. The value of <code>IsDefaultModel</code> can only be true for 1 model at a time.</p>
IsModelLocked	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the model is locked (<code>true</code>) or not (<code>false</code>). Records for locked models can only be added, updated, or deleted via the API.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description The language of the influence model.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The label for the influence model.</p>
ModelDescription	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The description of the influence model.</p>
ModelType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates whether the model is the default Salesforce influence model, or a custom model. These are the allowed picklist values.</p> <ul style="list-style-type: none"> • 1: Salesforce • 2: custom

CampaignMember

Represents the association between a campaign and either a lead or a contact.

Supported Calls

For API version 15.0 and earlier:`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

For API version 16.0 and later: `upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CampaignId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Campaign to which this Lead or Contact is associated.</p>
City	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The city for the address of the lead or contact.</p>
CompanyOrAccount	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The company or account of the lead or contact.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the Contact who is associated with a Campaign.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Restricted picklist, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The country for the address of the lead or contact.</p>
Description	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description The description of the associated lead or contact.</p>
DoNotCall	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates that the contact does not wish to be called.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Email address for the contact or lead.</p>
Fax	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Fax number for the contact or lead.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name of the contact or lead.</p>
FirstRespondedDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description
	Indicates the date on which the campaign member was first given a responded status.
HasOptedOutOfEmail	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the contact or lead would prefer not to receive email from Salesforce (<code>true</code>) or not (<code>false</code>).
HasOptedOutOfFax	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates that the contact or lead does not wish to receive faxes.
HasResponded	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the campaign member has responded to the campaign (<code>true</code>) or not (<code>false</code>). Label is Responded .
LastName	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The last name of the contact or lead.
LeadId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort
	Description
	Required. ID of the Lead who is associated with a Campaign.
LeadOrContactId	Type
	reference

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the associated lead or contact.</p>
LeadOrContactOwnerId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the owner of the associated lead or contact.</p>
LeadSource	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The source from which the lead was obtained.</p>
MobilePhone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The mobile phone number of the lead or contact.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description First and last name of the contact or lead with which the campaign member is associated.</p>
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The phone number of the lead or contact.</p>

Field	Details
PostalCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The postal code of the lead or contact.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object. To change the record type, modify the CampaignMemberRecordTypeId field on the associated Campaign.</p>
Salutation	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Salutation for the lead or contact.</p>
State	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The state for the address of the lead or contact.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Controls the HasResponded flag on this object. You can't directly set the HasResponded flag, as it is read-only, but you can set it indirectly by setting this field in a create or update call. Each predefined value implies a HasResponded flag value. Each time you update this field, you implicitly update the HasResponded flag. In the Salesforce user interface, Marketing users can define valid status values for the Status picklist. They can choose one status as the default status. For each Status field value, they can also select which values should be counted as "Responded," meaning that the HasResponded flag will be set to true for those values.</p>

Field	Details
	<p>40 character limit.</p> <p> Note: When creating or updating campaign members, use the text value for <code>Status</code> instead of the ID from the <code>CampaignMemberStatus</code> object.</p>
Street	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The street for the address of the lead or contact.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Title for the lead or contact.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates whether the campaign member is a lead or a contact.</p>



Note: If you are importing `CampaignMember` data into Salesforce and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Usage

Each record has a unique ID, and must contain either a `ContactId` or a `LeadId`, but can't contain both. Any attempt to create a single record with both results in a successful insert but only the `ContactId` will be inserted. However, you can create two separate records on a Campaign—one for the Lead and one for the Contact.

Standard fields from a Contact or Lead are associated with the `CampaignMember` object but you can't query them directly. To include a lead's `Phone` in your query, for example, query the field from the Lead object.

```
SELECT Id, (SELECT Phone FROM Lead)
FROM CampaignMember
```

This object is defined only for those organizations that have the marketing feature and valid marketing licenses. In addition, the object is accessible only to those users that are enabled as marketing users. If the organization does not have the marketing feature or valid

marketing licenses, this object does not appear in the `describeGlobal()` call, and you can't use `describeSObjects()` or `query()` with the CampaignMember object.

In API version 16.0 and later, a `create()` call only creates a new record; in earlier versions, a `create()` call creates and updates records. The API determines whether a record exists with the specified `CampaignId` and either `ContactId` or `LeadId`.



Note: Only use a `ContactId` or `LeadId`, but not both, unless you want to track lead-based campaign members you convert to contacts.

If the record does not exist for the given `ContactId` or `LeadId`, then a new record is created. If the record exists, an error is returned and no update is made. To update an existing record, specify the ID of the CampaignMember record to update.

In API versions 15.0 and earlier, if you submit multiple records using a single create request, and if more than one record matches an existing record, only the first record submitted updates the existing record. If any of the submitted records match each other but do not match existing records, only the last record submitted is created.

The `upsert()` call is not supported in API version 16.0 and later. To use the `upsert()` call on this object, you must first delete all data in ID fields except the record ID.

To delete a record, specify the ID of the CampaignMember record to delete.

When creating or updating records, the `Status` field value specified in the call is verified as a valid status for the given Campaign:

- If the specified `Status` value is a valid status, the value is updated, and the `HasResponded` field is updated to either `true` or `false`, depending on the `Status` value association with `HasResponded`.
- If the specified `Status` value is not a valid status, the API assigns the default status to the `Status` field and updates the `HasResponded` field with the associated value. However, if the given Campaign does not have a default status, the API assigns the value specified in the call to the `Status` field, and the `HasResponded` field is set to `false`.

SEE ALSO:

[Campaign](#)

[CampaignMemberStatus](#)

CampaignMemberStatus

One or more member status values defined for a campaign.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Customer Portal users can't access this object.

You can't delete a CampaignMemberStatus if that status is designated as the default status or if the status is currently used in a Campaign.

Fields

Field	Details
CampaignId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the campaign associated with this member status.</p>
HasResponded	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this status is equivalent to "Responded" (<code>true</code>) or not (<code>false</code>).</p>
IsDefault	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this status is the default status (<code>true</code>) or not (<code>false</code>).</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
Label	<p>Type string</p> <p>Properties Create, Filter, Sort</p> <p>Description Label for the status in the picklist. Limited to 765 characters.</p>
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
Description	
Order where this campaign member status appears in the picklist.	

Usage

Use this object to create picklist items for the member status in a campaign.

This object is defined only for those organizations that have the marketing feature and valid marketing licenses. In addition, the object is accessible only to those users that are enabled as marketing users. If the organization does not have the marketing feature or valid marketing licenses, this object does not appear in a `describeGlobal()` call, and you can't use `describeSObjects()` or `query()` with the CampaignMember object.

SEE ALSO:

[Campaign](#)

[CampaignMember](#)

CampaignOwnerSharingRule

Represents the rules for sharing a campaign with User records other than the owner or anyone above the owner in the role hierarchy.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CampaignAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group, or UserRole. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface. This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. A Campaign owned by a User in the source Group triggers the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p>

Field	Details
Properties	Create, Filter, Group, Sort
Description	The ID representing the User or Group being granted access.

Usage

Use this object to manage the sharing rules for campaigns.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

CampaignShare

Represents a sharing entry on a Campaign.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
CampaignId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Campaign associated with this sharing entry. This field can't be updated.</p>
CampaignAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to the Campaign. The possible values are:</p> <ul style="list-style-type: none"> • Read

Field	Details
	<ul style="list-style-type: none">• Edit• All (This value is not valid for creating or updating records.) <p>This field must be set to an access level that is higher than the organization's default access level for Campaign.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Valid values:</p> <ul style="list-style-type: none">• <code>Rule</code>—The User or Group has access via a Campaign sharing rule.• <code>Manual</code>—The User or Group has access because a User with “<code>All</code>” access manually shared the Campaign with them.• <code>Owner</code>—The User is the owner of the Campaign.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the Campaign. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit Campaign records owned by other users.

[CampaignTag](#)

Associates a word or short phrase with a Campaign.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

CampaignTag stores the relationship between its parent TagDefinition and the Campaign being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

Case

Represents a case, which is a customer issue or problem.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the account associated with this case.</p>
BusinessHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the business hours associated with this case.</p>
CaseNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description Assigned automatically when each case is inserted. It can't be set directly, and it can't be modified after the case is created.</p>
ClosedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	Description
	The date and time when the case was closed.
CommunityId	Type reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	ID of the Community (Zone) associated with this case.
	This field is available in API version 24.0 and later.
ConnectionReceivedId	Type reference
	Properties
	Filter, Nillable
	Description
	ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.
ConnectionSentId	Type reference
	Properties
	Filter, Nillable
	Description
	ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.
ContactEmail	Type email
	Properties
	Filter, Group, Nillable, Sort
	Description
	Email address for the Contact. The Case.ContactEmail field displays the Email field on the contact on page 495 that is referenced by Case.ContactId. Label is Contact Email. This field is available in API version 38.0 and later.
ContactFax	Type phone

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Fax number for the Contact. Label is <code>Contact Fax</code>. This field is available in API version 38.0 and later.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the associated Contact.</p>
ContactMobile	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Mobile telephone number for the Contact. Label is <code>Contact Mobile</code>. This field is available in API version 38.0 and later.</p>
ContactPhone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Telephone number for the Contact. Label is <code>Contact Phone</code>. This field is available in API version 38.0 and later.</p>
CreatorFullPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL of the user's profile photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
CreatorName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
Description	
CreatorSmallPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Name of the user who posted the question or reply. Only the first name of internal users (agents) appears to portal users in the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description A text description of the case. Limit: 32 KB.</p>
FeedItemId	<p>Type reference</p> <p>Properties Create, Group, Nullable, Sort</p> <p>Description ID of the question in Chatter associated with the case. This field is available in API version 33.0 and later, and is only accessible in organizations where Question-to-Case is enabled.</p>
HasCommentsUnreadByOwner	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a case contains comments that the case owner hasn't read (<code>true</code>) or not (<code>false</code>).</p>
HasSelfServiceComments	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	Description
	Indicates whether a case has comments added by a Self-Service user (<code>true</code>) or not (<code>false</code>).
IsClosed	Type boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the case is closed (<code>true</code>) or open (<code>false</code>). This field is controlled by the Status field; it can't be set directly. Label is <code>Closed</code> .
IsClosedOnCreate	Type boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the case was closed at the same time that it was created (<code>true</code>) or not (<code>false</code>). This flag is read-only and is automatically set when a record is created. It can't be set to <code>true</code> unless the IsClosed flag is also <code>true</code> .
IsDeleted	Type boolean
	Properties
	Defaulted on create, Filter
	Description
	Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is <code>Deleted</code> .
IsEscalated	Type boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates whether the case has been escalated (<code>true</code>) or not. A case's escalated state does not affect how you can use a case, or whether you can query, delete, or update it. You can set this flag via the API. Label is <code>Escalated</code> .
IsSelfServiceClosed	Type boolean
	Properties
	Defaulted on create, Filter, Group, Sort

Field	Details
Description	Indicates whether the case is closed for Self-Service users (<code>true</code>) or not (<code>false</code>).
IsStopped	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	Indicates whether an entitlement process on a case is stopped (<code>true</code>) or not (<code>false</code>).
IsVisibleInSelfService	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	Indicates whether the case can be viewed in the Customer Service Portal, Partner Service Portal, and Self-Service Portal (<code>true</code>) or not (<code>false</code>). This field is applied for case visibility in the Partner Relationship Management, Customer Service Portal, and the earlier version of Self Service Portal. The field does not alter sharing and will not prevent usage of a direct URL to a case if a portal user has read or write access.
LastReferencedDate	Type date
Properties	Filter, Nillable, Sort, Update
Description	The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date
Properties	Filter, Nillable, Sort, Update
Description	The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
Origin	Type picklist
Properties	Create, Filter, Group, Nillable, Sort, Update
Description	The source of the case, such as <code>Email</code> , <code>Phone</code> , or <code>Web</code> . Label is <code>Case Origin</code> .

Field	Details
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the contact who owns the case.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the parent case in the hierarchy. The label is Parent Case.</p>
Priority	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description The importance or urgency of the case, such as High, Medium, or Low.</p>
QuestionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The question in the answers community that is associated with the case. This field does not appear if you don't have an answers community enabled.</p>
Reason	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The reason why the case was created, such as Instructions not clear, or User didn't attend training.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p>

Field	Details
Description	ID of the record type assigned to this object.
SlaStartDate	Type dateTime
Properties	Create, Filter, Nillable, Update
Description	Shows the time that the case entered an entitlement process. If you have the "Edit" permission on cases, you can update or reset the time if you have the "Edit" permission on cases.
This field is available in API version 18.0 and later.	
Status	Type picklist
Properties	Create, Defaulted on create, Filter, Group, Nillable, Sort, Update
Description	The status of the case, such as "New," "Closed," or "Escalated." This field directly controls the <code>IsClosed</code> flag. Each predefined <code>Status</code> value implies an <code>IsClosed</code> flag value. For more information, see CaseStatus.
StopStartDate	Type dateTime
Properties	Filter, Nillable
Description	The date and time an entitlement process was stopped on the case.
This field is available in API version 18.0 and later.	
Subject	Type string
Properties	Create, Filter, Group, Nillable, Sort, Update
Description	The subject of the case. Limit: 255 characters.
SuppliedCompany	Type string
Properties	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	<p>Description The company name that was entered when the case was created. This field can't be updated after the case has been created. Label is <code>Company</code>.</p>
SuppliedEmail	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The email address that was entered when the case was created. This field can't be updated after the case has been created. Label is <code>Email</code>. If your organization has an active auto-response rule, <code>SuppliedEmail</code> is required when creating a case via the API. Auto-response rules use the email in the contact specified by <code>ContactId</code>. If no email address is in the contact record, the email specified here is used.</p>
SuppliedName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The name that was entered when the case was created. This field can't be updated after the case has been created. Label is <code>Name</code>.</p>
SuppliedPhone	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The phone number that was entered when the case was created. This field can't be updated after the case has been created. Label is <code>Phone</code>.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The type of case, such as <code>Feature Request</code> or <code>Question</code>.</p>



Note: If you are importing Case data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself.

Usage

Use the Case object to manage cases for your organization. Client applications can query, update, and delete Attachment records associated with a case via the API.

Assignment Rules

When you query or update a case, your client application can have the case automatically assigned to one or more User records based on assignment rules that have been configured in the user interface. To use this feature, your client application must set either of the following options (but not both) in the AssignmentRuleHeader used in the create or update:

Field	Field Type	Details
assignmentRuleId	reference	ID of the assignment rule to use. Can be an inactive assignment rule. If unspecified and <code>useDefaultRule</code> is <code>true</code> , then the default assignment rule is used. To find the ID for a given assignment rule, query the AssignmentRule object (specifying <code>RuleType="caseAssignment"</code>), iterate through the returned AssignmentRule objects, find the one you want to use, retrieve its ID, and then specify its ID in this field in the AssignmentRuleHeader.
useDefaultRule	boolean	Specifies whether to use the default rule for rule-based assignment (<code>true</code>) or not (<code>false</code>). The default rule is assigned by users in the Salesforce user interface.

For a code example that shows setting the AssignmentRuleHeader for a Lead (which is similar to setting the AssignmentRuleHeader for a Case), see [Lead](#).

Separating Accounts from Contacts in Cases

In releases before 8.0, the `AccountId` could not be specified; it was derived from the contact's account. This behavior will continue to be supported in future releases, but you can also now specify an `AccountId`. If you do not specify the `AccountId` during the creation of a case, the value will default to the contact's `AccountId`.

 **Note:** When a record is updated, if the `ContactId` has not changed, then the `AccountId` is not regenerated. This prevents the API from overwriting a value previously changed in the Salesforce user interface. However, if an API call changes the `ContactId` and the `AccountId` field is empty, then the `AccountId` is generated using the contact's account.

Using `_case` with Java

Depending on the development tool you use, you might need to write your application using `_case` instead of `case`, because `case` is a reserved word in Java.

SEE ALSO:

[Account](#)

[CaseMilestone](#)

CaseArticle

Represents the association between a Case and a KnowledgeArticle. This object is available in API version 20.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Access to this object is controlled by the parent Case and KnowledgeArticle. However, when querying, access is only controlled by the parent Case.

Customer Portal users can't access this object.

Fields

Field	Details
ArticleLanguage	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The language of the article associated with the case.</p>
ArticleVersionNumber	<p>Type int</p> <p>Properties Create, Group, Nullable</p> <p>Description The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>
CaseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the Case associated with the KnowledgeArticle.</p>
IsSharedByEmail	<p>Type int</p>

Field	Details
	<p>Properties Create, Group, Nillable</p> <p>Description Indicates that the article has been shared with the customer through an email.</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the KnowledgeArticle associated with the Case.</p>

Usage

This object represents the association of a knowledge article with a Case. An article is associated with a case when it's relevant to a specific issue, when it helps an agent solve the case, or when the agent sends the article to a customer.

You can use this object to include case-article associations in Apex and Visualforce.

You can't update this object via the API. If you attempt to create a record that matches an existing record, the create request simply returns the existing record.

SEE ALSO:

[Case](#)
[KnowledgeArticle](#)

CaseComment

Represents a comment that provides additional information about the associated Case.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()`

Fields

Field	Details
CommentBody	<p>Type textarea</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Text of the CaseComment. The maximum size of the comment body is 4,000 bytes. Label is Body.</p>
ConnectionReceivedId	<p>Type reference</p>
	<p>Properties Filter, Nillable</p>
	<p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p>
	<p>Properties Filter, Nillable</p>
	<p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
CreatorFullPhotoUrl	<p>Type string</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description URL of the user's profile photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
CreatorName	<p>Type string</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description Name of the user who posted the question or reply. Only the first name of internal users (agents) appears to portal users in the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>

Field	Details
CreatorSmallPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL of the user's thumbnail photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPublished	<p>Type boolean</p> <p>Properties Defaulted on create, Create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the CaseComment is visible to customers in the Self-Service portal (<code>true</code>) or not (<code>false</code>). Label is Published. This is the only CaseComment field that can be updated via the API.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort,</p> <p>Description Required. ID of the parent Case of the CaseComment.</p>



Note: If you are importing CaseComment data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Usage

In the Salesforce user interface, comments are generally entered by a User working on a Case. All users have access to create and view CaseComment in the Salesforce user interface and when using the API. In the API, CaseComment records can't be modified after insertion

unless the user has the "Modify All" object-level permission for Cases or the "Modify All Data" permission. If not, users can only update the `IsPublished` field, and can't delete CaseComment.

SEE ALSO:[Object Basics](#)

CaseContactRole

Represents the role that a given Contact plays on a Case.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CasesId	Type reference Properties Create, Filter, Group, Sort Description ID of the cases associated with this contact.
ContactId	Type reference Properties Create, Filter, Group, Sort, Update Description Required. ID of the contact.
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
Role	Type picklist

Field	Details
Properties	Create, Filter, Group, Nillable, Sort, Update
Description	Name of the role played by the contact on this contract, such as Decision Maker, Approver, Buyer, and so on. Must be unique—there can't be multiple records in which the ContractId, ContactId, and Role values are identical. Different contacts can play the same role on the same contract. A contact can play different roles on the same contract.

Usage

Use this object to define the role that a given Case plays on a given Contact. For example, you can use this object to be able to see all contacts who are associated to a case, or, given a contact, be able to query all cases that they are associated with, even if they are not the primary contact on the case.

CaseFeed

Represents a single feed item in the feed displayed on the detail page for a case record. A case feed shows recent changes to a case record for any fields that are tracked in feeds, and comments and posts about the record. It is a useful way to stay up-to-date with changes to cases. This object is available in API version 18.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Case object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	Type textarea Properties Nillable, Sort

Field	Details
Description	
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission. This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	
ContentData	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator and the <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentDescription	
ContentDescription	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if <code>Type</code> is <code>ContentPost</code>. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. Use <code>FeedItem</code> instead. The ID of the associated FeedPost. A FeedPost represents the following types of changes in a CaseFeed: status updates, changes to tracked fields, text posts, link posts, and content posts.</p>

Field	Details
	<p>Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
<code>InsertedById</code>	<p>Type reference</p> <p>Properties Group, Nullable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
<code>IsDeleted</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
<code>IsRichText</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none">• <code><p></code> <p class="list-item-l1">•  Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none">• <code></code>• <code><code></code>• <code><i></code>• <code><u></code>• <code><s></code>• <code></code>• <code></code>• <code></code>

Field	Details
	<ul style="list-style-type: none">• <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a <code>FeedComment</code> is inserted on the feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for the <code>FeedComment</code>. Deleting the <code>FeedComment</code> does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of <code>FeedLikes</code> associated with this feed item.</p>
LinkURL	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of the <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none">• <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.• <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>.• For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>.• You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the case record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordID	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion article associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a CaseFeed object of Type ContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The title of the CaseFeed. When the <code>Type</code> is LinkPost, the <code>LinkURL</code> is the URL, and this field is the link name.</p>

Field	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of CaseFeed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence. • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p>

Field	Details
	<ul style="list-style-type: none">• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case.• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case.• <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case.• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received.• <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated.• <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status.• <code>SocialPost</code>—generated when a social post is created from a case.
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• <code>Visibility</code> can be updated on record posts.• The <code>Update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for a case record.

SEE ALSO:

[Case](#)
[EntitySubscription](#)
[NewsFeed](#)
[FeedItem](#)
[UserProfileFeed](#)

CaseHistory

Represents historical information about changes that have been made to the associated Case.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is always read-only.

Fields

Field	Details
CaseId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Case associated with this record.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Name of the case field that was modified, or a special value to indicate some other modification to the case. The possible values, in addition to the case field names, are:</p> <ul style="list-style-type: none">• ownerAssignment—The owner of the case was changed.• ownerAccepted—A user took ownership of a case from a queue.

Field	Details
	<ul style="list-style-type: none"> ownerEscalated—The owner of the case was changed due to case escalation. external—A user made the case visible to customers in the Customer Self-Service Portal.
IsDeleted	Type boolean
	Properties Defaulted on create, Filter
	Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
NewValue	Type anyType
	Properties Nillable, Sort
	Description New value of the modified case field. Maximum of 255 characters.
OldValue	Type anyType
	Properties Nillable, Sort
	Description Previous value of the modified case field. Maximum of 255 characters.

Usage

Case history entries are indirectly created each time a case is modified.

Two rows are added to this record when foreign key fields change. One row contains the foreign key object names that display in the online application. For example, Jane Doe is recorded as the name of a Contact. The other row contains the actual foreign key ID that is only returned to and visible from the API.

This object respects field level security on the parent object.

SEE ALSO:

[Object Basics](#)

CaseMilestone

Represents a milestone (required step in a customer support process) on a Case. This object is available in API version 18.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve(), update()`

Fields

Field	Details
BusinessHoursId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the BusinessHours associated with the CaseMilestone.</p>
CaseId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the case.</p>
CompletionDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Update</p> <p>Description The date and time the milestone was completed.</p>
ElapsedTimeInDays	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description The time required to complete a milestone in days.</p>
ElapsedTimeInHrs	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description The time required to complete a milestone in hours.</p>

Field	Details
Elapsed Time In Mins	Type int Properties Filter, Nillable Description The time required to complete a milestone in minutes.
Is Completed	Type boolean Properties Defaulted on create, Filter Description Indicates whether the milestone is completed (<code>true</code>) or not (<code>false</code>).
Is Violated	Type boolean Properties Defaulted on create, Filter Description Indicates whether the milestone is violated (<code>true</code>) or not (<code>false</code>).
Milestone Type Id	Type reference Properties Filter, Nillable Description The ID of the milestone on the case.
Start Date	Type dateTime Properties Filter, Nillable, Update Description The date and time the milestone started on the case.
Target Date	Type dateTime Properties Filter Description The date and time the milestone must be completed.

Field	Details
TargetResponseInDays	Type double
	Properties Filter, Nillable
	Description
	The time to complete the milestone in days.
TargetResponseInHrs	Type double
	Properties Filter, Nillable
	Description
	The time to complete the milestone in hours.
TargetResponseInMins	Type int
	Properties Filter, Nillable
	Description
	The time to complete the milestone in minutes.
TimeSinceTargetInMins	Type text
	Properties Nillable
	Description
	The time elapsed since the milestone target. The format is minutes and seconds.
TimeRemainingInMins	Type text
	Properties Nillable
	Description
	Time remaining to reach the milestone target. The format is minutes and seconds.

Usage

This object lets you view a milestone on a case. It also lets you view if the milestone was completed and when it must be completed.

SEE ALSO:

[Case](#)
[MilestoneType](#)
[SlaProcess](#)

CaseOwnerSharingRule

Represents the rules for sharing a case with users other than the owner.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Cases owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field	Details
	<p>Description The ID representing the target user or group. Target users or groups are given access.</p>

Usage

Use this object to manage the sharing rules for cases. General sharing and territory management-related sharing use this object.

SEE ALSO:

[Case](#)

[CaseShare](#)

[Metadata API Developer Guide: SharingRules](#)

CaseShare

Represents a sharing entry on a Case.

Supported Calls

`describeSObjects(), create(), delete(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
CaseAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to the Case. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All This value is not valid for creating or deleting records.

Field	Details
CaseId	<p>This field must be set to an access level that is higher than the organization's default access level for cases.</p>
IsDeleted	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Case associated with this sharing entry. This field can't be updated.</p> <p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none"> • <code>Manual</code>—The User or Group has access because a user with “All” access manually shared the Case with them. • <code>Owner</code>—The User is the owner of the Case. • <code>ImplicitChild</code>—User or Group has access to the Case on the Account associated with this Case. • <code>Rule</code>—The User or Group has access via a Case sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the Case. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view and edit Case records owned by other users.

If you attempt to create a new record that matches an existing record, request updates any modified fields and returns the existing record.

SEE ALSO:

[AccountShare](#)
[LeadShare](#)
[OpportunityShare](#)

CaseSolution

Represents the association between a Case and a Solution.

Supported Calls

`create(), delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
CaseId	Type reference Properties Create, Filter, Group, Sort Description Required. ID of the Case associated with the Solution.
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
SolutionId	Type reference Properties Create, Filter, Group, Sort

Field	Details
	<p>Description Required. ID of the Solution associated with the case.</p>

Usage

You can't update this object via the API. If you attempt to create a record that matches an existing record, the request simply returns the existing record.

SEE ALSO:

[CaseShare](#)

[SolutionStatus](#)

CaseStatus

Represents the status of a Case, such as New, On Hold, or In Process.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this case status value represents a closed Case (<code>true</code>) or not (<code>false</code>). Multiple case status values can represent a closed Case.</p>
IsDefault	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the default case status value (<code>true</code>) or not (<code>false</code>) in the picklist.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this case status value. This display value is the internal label that does not get translated.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the case status picklist. These numbers are not guaranteed to be sequential, as some previous case status values might have been deleted.</p>

Usage

This object represents a value in the case status picklist. The case status picklist provides additional information about the status of a Case, such as whether a given `Status` value represents an open or closed case. Query the `CaseStatus` object to retrieve the set of values in the case status picklist, and then use that information while processing Case records to determine more information about a given case. For example, the application could test whether a given case is open or closed based on its `Status` value and the value of the `IsClosed` property in the associated `CaseStatus` object.

SEE ALSO:

[Object Basics](#)

CaseTag

Associates a word or short phrase with a Case

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

CaseTag stores the relationship between its parent TagDefinition and the Case being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

CaseTeamMember

Represents a case team member, who works with a team of other users to help resolve a case.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Fields

Field	Details
MemberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user or contact who is a member on a case team.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the case with which the case team member is associated.</p>
TeamRoleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the case team role with which the case team member is associated.</p>
TeamTemplateMemberId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the team member included in a predefined case team.</p>

CaseTeamRole

Represents a case team role. Every case team member has a role on a case, such as "Customer Contact" or "Case Manager."

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group for cases. The possible values are:</p> <ul style="list-style-type: none">• None• Read• Edit
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the case team role.</p>
PreferencesVisibleInCSP	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether or not the case team role is visible to Customer Portal users.</p>

CaseTeamTemplate

Represents a predefined case team, which is a group of users that helps resolve a case.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A text description of the predefined case team.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the predefined case team.</p>

CaseTeamTemplateMember

Represents a member on a predefined case team, which is a group of users that helps resolve cases.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
MemberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user or contact who is a team member on a predefined case team.</p>

Field	Details
TeamRoleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the predefined case team member's case team role.</p>
TeamTemplateId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the predefined case team's template.</p>

CaseTeamTemplateRecord

The CaseTeamTemplateRecord object is a linking object between the Case and CaseTeamTemplate objects. To assign a predefined case team to a case (customer inquiry), create a CaseTeamTemplateRecord record and point the ParentId to the case and the TeamTemplateId to the predefined case team.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the case with which the case team template record is associated.</p>
TeamTemplateId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field	Details
	<p>Description The ID of the predefined case team with which the case team template record is associated.</p>

CategoryData

Represents a logical grouping of Solution records.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CategoryNodeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the CategoryNode associated with the solution.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
RelatedObjectId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the solution related to the category.</p>

Usage

This object allows you to assign one or more categories to a Solution. It is an intermediate data table with two foreign keys that defines the relationship between a CategoryNode and a Solution record.

CategoryData has two foreign keys:

- The first foreign key, `CategoryNodeId`, refers to the ID of a CategoryNode.
- The other foreign key, `RelatedSobjectId`, refers to a Solution ID.

This is a many-to-many relationship, so there can be multiple rows returned with a `CategoryNodeId`. A Solution can be associated with multiple categories.

SEE ALSO:

[Object Basics](#)

CategoryNode

Represents a tree of Solution categories.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

- Customer Portal users can't access this object.
- Attempting to delete a CategoryNode that has children (referred by `CategoryNode.Parent`), or is referred to elsewhere, causes a failure.

Fields

Field	Details
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label for the category node.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	Description
	ID of the parent of this node, if any.
SortOrder	Type int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Indicates the sort order of child CategoryNode objects.
SortStyle	Type picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	Indicates whether the sort order is alphabetical or custom.

Usage

A CategoryNode defines a category of solutions. In the user interface, you can edit category definitions from Setup by entering *Solution Categories* in the Quick Find box, then selecting **Solution Categories**.

SEE ALSO:

[CategoryData](#)

[Solution](#)

CategoryNodeLocalization

When the Translation Workbench is enabled for your organization, the CategoryNodeLocalization object provides the translation of the label of a solution category.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- Your organization must be using Professional, Enterprise, Developer, Unlimited, or Performance Edition and be enabled for the Translation Workbench.
- To view this object, you must have the “View Setup and Configuration” permission.

Fields

Field	Details
CategoryNodeId	<p>Type reference</p> <p>Properties Create, Filter, Nullable</p> <p>Description The ID of the solution CategoryNode that is being translated.</p>
LanguageLocaleKey	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Restricted picklist</p> <p>Description This field is available in API version 16.0 and earlier. It is the same as the <code>Language</code> field.</p>
Language	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Restricted picklist</p> <p>Description This field is available in API version 17.0 and later. The combined language and locale ISO code, which controls the language for labels displayed in an application. This picklist contains the following fully-supported languages:</p> <ul style="list-style-type: none"> • Chinese (Simplified): <code>zh_CN</code> • Chinese (Traditional): <code>zh_TW</code> • Danish: <code>da</code> • Dutch: <code>nl_NL</code> • English: <code>en_US</code> • Finnish: <code>fi</code> • French: <code>fr</code> • German: <code>de</code> • Italian: <code>it</code> • Japanese: <code>ja</code> • Korean: <code>ko</code> • Norwegian: <code>no</code> • Portuguese (Brazil): <code>pt_BR</code> • Russian: <code>ru</code>

Field	Details
	<ul style="list-style-type: none">• Spanish: es• Spanish (Mexico): es_MX• Swedish: sv• Thai: th <p>The following end-user only languages are available.</p> <ul style="list-style-type: none">• Arabic: ar• Bulgarian: bg• Croatian: hr• Czech: cs• English (UK): en_GB• Greek: el• Hebrew: iw• Hungarian: hu• Indonesian: in• Polish: pl• Portuguese (Portugal): pt_PT• Romanian: ro• Slovak: sk• Slovenian: sl• Turkish: tr• Ukrainian: uk• Vietnamese: vi
	<p>The following platform languages are available for organizations that use Salesforce exclusively as a platform.</p> <ul style="list-style-type: none">• Albanian: sq• Arabic (Algeria): ar_DZ• Arabic (Bahrain): ar_BH• Arabic (Egypt): ar_EG• Arabic (Iraq): ar_IQ• Arabic (Jordan): ar_JO• Arabic (Kuwait): ar_KW• Arabic (Lebanon): ar_LB• Arabic (Libya): ar LY• Arabic (Morocco): ar_MA• Arabic (Oman): ar_OM• Arabic (Qatar): ar_QA• Arabic (Saudi Arabia): ar_SA• Arabic (Sudan): ar_SD

Field	Details
	<ul style="list-style-type: none">Arabic (Syria): ar_SYArabic (Tunisia): ar_TNArabic (United Arab Emirates): ar_AEArabic (Yemen): ar_YEArmenian: hyBasque: euBosnian: bsBengali: bnChinese (Simplified—Singapore): zh_SGChinese (Traditional—Hong Kong): zh_HKEnglish (Australia): en_AUEnglish (Canada): en_CAEnglish (Hong Kong): en_HKEnglish (India): en_INEnglish (Ireland): en_IEEnglish (Malaysia): en_MYEnglish (Philippines): en_PHEnglish (Singapore): en_SGEnglish (South Africa): en_ZAEstonian: etFrench (Belgium): fr_BEFrench (Canada): fr_CAFrench (Luxembourg): fr_LUFrench (Switzerland): fr_CHGeorgian: kaGerman (Austria): de_ATGerman (Luxembourg): de_LUGerman (Switzerland): de_CHHindi: hiIcelandic: isIrish: gaItalian (Switzerland): it_CHLatvian: lvLithuanian: ltLuxembourgish: lbMacedonian: mkMalay: msMaltese: mt

Field	Details
	<ul style="list-style-type: none"> • Romanian (Moldova): <code>ro_MD</code> • Montenegrin: <code>sh_ME</code> • Romansh: <code>rm</code> • Serbian (Cyrillic): <code>sr</code> • Serbian (Latin): <code>sh</code> • Spanish (Argentina): <code>es_AR</code> • Spanish (Bolivia): <code>es_BO</code> • Spanish (Chile): <code>es_CL</code> • Spanish (Colombia): <code>es_CO</code> • Spanish (Costa Rica): <code>es_CR</code> • Spanish (Dominican Republic): <code>es_DO</code> • Spanish (Ecuador): <code>es_EC</code> • Spanish (El Salvador): <code>es_SV</code> • Spanish (Guatemala): <code>es_GT</code> • Spanish (Honduras): <code>es_HN</code> • Spanish (Nicaragua): <code>es_NI</code> • Spanish (Panama): <code>es_PA</code> • Spanish (Paraguay): <code>es_PY</code> • Spanish (Peru): <code>es_PE</code> • Spanish (Puerto Rico): <code>es_PR</code> • Spanish (United States): <code>es_US</code> • Spanish (Uruguay): <code>es_UY</code> • Spanish (Venezuela): <code>es_VE</code> • Tagalog: <code>tl</code> • Tamil: <code>ta</code> • Urdu: <code>ur</code> • Welsh: <code>cy</code> <p>The values in this field are not related to the default locale selection.</p>

NamespacePrefix	Type string Properties Filter, Nullable Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix_componentName</code> notation. The namespace prefix can have one of the following values:
-----------------	--

Field	Details
	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Value	<p>Type string</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The actual translated label for the solution category. Label is Translation.</p>

Usage

Use this object to translate the labels of your solution categories into a supported language. Users with the Translation Workbench enabled can view category node translations, but either the “Customize Application,” “Manage Translation,” or “Manage Categories” permission is required to create or update category node translations.

SEE ALSO:

[ScontrolLocalization](#)

[WebLinkLocalization](#)

ChatterActivity

ChatterActivity represents the number of posts and comments made by a user and the number of comments and likes on posts and comments received by the same user. This object is available in API version 23.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
CommentCount	<p>Type int</p>

Field Name	Details
	Properties Filter, Group, Sort
CommentReceivedCount	Description The number of FeedComments made by the ParentId.
InfluenceRawRank	Type int
	Properties Filter, Group, Sort
LikeReceivedCount	Description The number of FeedLikes received by the ParentId.
NetworkId	Type reference
	Properties Filter, Group, Nillable, Sort
ParentId	Description Identifier of the community to which the ChatterActivity belongs. This field is available only if Salesforce Communities is enabled in your organization. This field is available in API version 26.0 and later.
	Type reference
	Properties Filter, Group, Nillable, Sort
	Description The number of FeedComments made by the ParentId.
	Type int
	Properties Filter, Group, Sort
	Description The number of FeedLikes received by the ParentId.
	Type reference
	Properties Filter, Group, Nillable, Sort
	Description Identifier of the community to which the ChatterActivity belongs. This field is available only if Salesforce Communities is enabled in your organization. This field is available in API version 26.0 and later.
	Type reference
	Properties Filter, Group, Nillable, Sort

Field Name	Details
	<p>Description Identifier of the object type to which the ChatterActivity is related. In API version 38.0, the ParentId must be a UserId or SelfServiceUser ID.</p>
PostCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedItems made by the ParentId.</p>

Usage

- Use this object to reference the Chatter activity statistics, which include the number of posts and comments made by a user and the number of comments and likes on posts and comments received by the same user.
- You can directly query for ChatterActivity.

```
SELECT Id, PostCount, LikeReceivedCount
FROM ChatterActivity
WHERE ParentId = UserId
```



Note: To query ChatterActivity, you must provide the ParentId. In API version 38.0, the ParentId must be a UserId or SelfServiceUser ID.

- A ChatterActivity record is created for users the first time they post or comment. Users who have never posted or commented don't have ChatterActivity records. If users make only one post and then delete it, they do have ChatterActivity records. In both cases, the user interface displays zeros for their Chatter activity.
- Use the InfluenceRawRank field to reference a user's Chatter influence rank. This field is available in API version 26.0 and later.

SEE ALSO:

[FeedItem](#)
[FeedComment](#)
[FeedLike](#)

ChatterAnswersActivity

Represents the reputation of a User in Chatter Answers communities. This object is available in API version 25.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
BestAnswerReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of best answers the User has received from other users.</p>
BestAnswerSelectedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of best answers the User has selected.</p>
QuestionsCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of Question records posted by the User.</p>
QuestionSubscrCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of Question records the User has selected to follow.</p>
QuestionSubscrReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of users following Question records posted by the User.</p>
QuestionUpVotesCount	<p>Type int</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of up votes the User has marked on Question records posted by other users.</p>
QuestionUpVotesReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of up votes the User has received from other users on the Question records he or she has posted.</p>
RepliesCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of Reply records posted by the User.</p>
ReplyDownVotesCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of down votes the User has marked on Reply records posted by other users.</p>
ReplyDownVotesReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of down votes the User has received from other users on the Reply records he or she has posted.</p>
ReplyUpVotesCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	The number of up votes the User has marked on the Reply records posted by other users.
ReplyUpVotesReceivedCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The number of up votes the User has received from other users on the Reply records he or she has posted.
ReportAbuseOnQuestionsCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The number of abuses that the User has reported on Question records posted by other users.
ReportAbuseOnRepliesCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The number of abuses that the User has reported on Reply records posted by other users.
ReportAbuseReceivedOnQnCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The number of abuses reported by other users on the Question records posted by the User.
ReportAbuseReceivedOnReCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort

Field Name	Details
	<p>Description the number of abuses reported by other users on the Reply records posted by the User.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The User ID associated with this reputation.</p>
CommunityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID for the zone associated with this reputation.</p>

Usage

Use this object to view metrics on User activity in Chatter Answers. For example, you can use the ChatterAnswersActivity object to view the number of Question records a user is following in Chatter Answers communities.

SEE ALSO:

[Question](#)
[Reply](#)
[User](#)

ChatterAnswersReputationLevel

Represents a reputation level within a Chatter Answers zone. This object is available in API version 26.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
CommunityID	Type reference Properties Create, Filter, Group, Sort Description ID of the zone for which you're creating the reputation level.
Name	Type string Properties Create, Filter, Group, Sort, Update Description Name of the reputation level.
Value	Type int Properties Create, Filter, Group, Sort, Update Description Minimum number of points for this level.

Usage

Use to create or edit reputation levels for the zone.

ChatterConversation

Represents a private conversation in Chatter, consisting of messages that conversation members have sent or received. This object is available in API version 23.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the conversation.</p>

Usage

Use this object to identify private conversations in Chatter. Users can access this object if they have the “Manage Chatter Messages” permission. This object is read-only via the API and is provided only to allow administrators to view users’ Chatter messages; for example, for compliance purposes.

SEE ALSO:

[ChatterConversationMember](#)
[ChatterMessage](#)

ChatterConversationMember

Represents a member of a private conversation in Chatter. A member has either sent messages to or received messages from other conversation participants. This object is available in API version 23.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
ConversationId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the associated ChatterConversation.</p>
MemberId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p>
	<p>Description ID of the conversation member.</p>

Usage

Use this object to view members of private conversations in Chatter. Users can access this object if they have the "Manage Chatter Messages" permission. This object is read-only via the API and is provided only to allow administrators to view users' Chatter messages; for example, for compliance purposes.

SEE ALSO:

[ChatterConversation](#)

[ChatterMessage](#)

ChatterMessage

Represents a message sent as part of a private conversation in Chatter. This object is available in API version 23.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties None</p> <p>Description Text of the message.</p>
ConversationId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the conversation that the ChatterMessage is associated with.</p>

Field Name	Details
SenderId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the sender.</p>
SenderNetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the community from which the message was sent. This field is available only if Salesforce Communities is enabled in your organization. This field is available in API version 32.0 and later.</p>
SentDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description Date the message was sent.</p>

Usage

Use this object to view and delete messages sent or received via private conversations in Chatter. Users can access this object if they have the “Manage Chatter Messages” permission. Users with the “Moderate Communities Chatter Messages” permission can access this object in communities they’re a member of, only if the message has been flagged as inappropriate. This object is provided to allow administrators to view and delete users’ Chatter messages, for example, for compliance purposes.

Messages are hard deleted. They aren’t sent to the recycle bin.

Deleting a message that resulted from sharing a file with someone does not delete the file itself.

SEE ALSO:

- [ChatterConversation](#)
- [ChatterConversationMember](#)

ClientBrowser

Represents a cookie added to the browser upon login, and also includes information about the browser application where the cookie was inserted. This object is available in version 28.0 and later.

Supported Calls

`describeSObjects(), delete(), query(), retrieve()`

Fields

Field	Details
FullUserAgent	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Detailed information about the client (browser). For example, Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.1) Gecko/2008070208 Firefox/3.0.1</p>
LastUpdate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Represents the last time the cookie was changed.</p>
ProxyInfo	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The browser's current proxy information.</p>
UserId	<p>Type reference</p> <p>Properties Nullable, Group, Sort</p> <p>Description The ID of the user associated with this item.</p>

Usage

At every login, the device the login request is from is checked against the known devices using ClientBrowser. A match means a cookie was found on the browser that matches an entry in the ClientBrowser table, so the device is known. No match means that no matching cookie was found, so the device is unknown, and the user is asked to confirm their identity.

CollaborationGroup

Represents a Chatter group. This object is available in API version 19.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

The visibility of information in groups depends on the type of group and the user's permissions.

- **Members:** Any user with the "Create and Own New Chatter Groups" permission can create public, private, and unlisted groups, including in any communities they belong to.
- **Owners and managers:** Users can modify group details for any group they own or manage. Owners can also delete groups they own.
- **Nonmembers:** These user permissions allow group access regardless of group membership.
 - "View All Data"—Allows users to view all public and private groups across their organization and its communities. Users with this permission can't view unlisted group information, unless they have the "Modify Unlisted Groups" permission as well.
 - "Modify All Data"—Allows users to view, modify, and delete all public and private groups across their organization and its communities. Users with this permission can't view or modify unlisted group information, unless they have the "Manage Unlisted Groups" permission as well.
 - "Create and Customize Communities"—Allows users to view, modify, and delete all public and private groups in communities.
 - "Manage Unlisted Groups"—Allows users to search for, access, and modify any unlisted group in an organization and its communities.
 - "Data Export"—Allows users to export any data from Salesforce, including private and unlisted group data from an organization and its communities.
- **Apex and Visualforce:** Apex code runs in system mode, which means that the permissions of the current user aren't taken into account.
 - Visualforce pages that display groups might expose unlisted or private group data to users who aren't members.
 - Because system mode disregards the user's permissions, all users who are accessing a Visualforce page that's showing a group can act as an owner of that group.
 - AppExchange apps that are written in Apex and that access all groups will expose unlisted groups to users who aren't members.

To limit and manage access to the unlisted and private groups in your organization:

- Explicitly filter out unlisted and private group information from SOQL queries in all Apex code.
- Use permission sets, profile-level permissions, and sharing checks in your code to further limit group access.
- Use Apex triggers on the CollaborationGroup object to monitor and manage the creation of groups. In Setup, enter *Group Triggers* in the Quick Find box, then select **Group Triggers** to add triggers.

Fields

Field	Details
AnnouncementId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Contains the ID of the Announcement last associated with the group. This field is available in API version 30.0 and later.</p>
BannerPhotoUrl	<p>Type url</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The URL for the group's banner photo. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo. This field is available in API version 36.0 and later.</p>
CanHaveGuests	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If set to <code>true</code>, indicates that a group allows customers. Chatter customers are users outside your company's email domains who can only see groups they're invited to and interact with members of those groups; they can't see any Salesforce information. This field is available starting in API version 23.0, but groups that allow customers are accessible from earlier API versions. However, when accessed from earlier API versions, groups that allow customers aren't distinguishable from private groups. We strongly recommend that you upgrade to the latest API version. If you must use an earlier version, name groups that allow customers to indicate that they include customers.</p>
CollaborationType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of Chatter group. Available values are:</p> <ul style="list-style-type: none"> • <code>Public</code>—Anyone can see and post updates. Anyone can join a public group.

Field	Details
	<ul style="list-style-type: none"> • Private—Only members can see the group feed and post updates. Non-members can only see the group name and a few other details in list views, search, and on the group page. The group's owner or managers must add members who request to join the group. • Unlisted—Only members and users with the "Manage Unlisted Groups" permission can see the group and post updates. Other users can't access the group or see it in lists, search, and feeds.
Description	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the group.</p>
FullPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the group's profile photo. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo. This field is available in API version 20.0 and later.</p>
GroupEmail	<p>Type email</p> <p>Properties Group, Nillable, Sort</p> <p>Description The email address for posting to the group. For private groups, only visible to members and users with "Modify All Data" or "View All Data" permissions. This field is available in API version 29.0 and later.</p>
HasPrivateFieldsAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	Description
	<p>If set to <code>true</code>, indicates that a user can see the <code>InformationBody</code> and <code>InformationTitle</code> fields in a private group. This field is set to <code>true</code> for members of a private group and users with "Modify All Data" or "View All Data" permissions.</p>
InformationBody	Type
	textarea
	Properties
	Create, Nullable, Update
	Description
	<p>The text of the Information section. For private groups, only visible to members and users with "Modify All Data" or "View All Data" permissions.</p>
InformationTitle	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	<p>The title of the Information section. For private groups, only visible to members and users with "Modify All Data" or "View All Data" permissions.</p>
IsArchived	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	<p>Indicates whether the group is archived (<code>true</code>) or not (<code>false</code>).</p>
IsAutoArchiveDisabled	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	<p>Indicates whether automatic archiving is disabled for the group (<code>true</code>) or not (<code>false</code>).</p>
IsBroadcast	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	<p>Indicates whether the group is a broadcast group (<code>true</code>) or not (<code>false</code>).</p>
	<p>This field is available in API version 36.0 and later.</p>

Field	Details
LastFeedModifiedDate	Type dateTime
	Properties Filter, Sort
	Description The date of the last post or comment on the group.
LastReferencedDate	Type date
	Properties Filter, Nillable, Sort, Update
	Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date
	Properties Filter, Nillable, Sort, Update
	Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
MediumPhotoUrl	Type url
	Properties Filter Nillable Sort
	Description The URL for the larger, cropped photo size.
MemberCount	Type int
	Properties Filter, Group, Nillable, Sort
	Description The number of members in the group.
Name	Type string

Field	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the group. Group names must be unique across public and private groups. Unlisted groups don't require unique names.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the Community that this group is part of. This field is available only if Salesforce Communities is enabled in your organization. You can only add a <code>NetworkId</code> when creating a group. You can't change or add a <code>NetworkId</code> for an existing group. This field is available in API version 26.0 and later.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the group. Only the current group owner or people with the "Modify All Data" permission can update the <code>OwnerId</code>.</p>
SmallPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for a thumbnail of the group's profile photo. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo. This field is available in API version 20.0 and later.</p>

Usage

Use this object to create, edit, or delete groups in an organization or in a community. Deleting a group permanently deletes all posts and comments to the group. It also deletes all files and links posted to the group and removes the files from other locations where they were shared.

As a Chatter group member, you can post to the group using the CollaborationGroupFeed object. As a Chatter group owner or manager, you can add or remove group members using the CollaborationGroupMember object, post announcements to the group using the Announcement object, and accept or decline requests to join private groups using the CollaborationGroupMemberRequest object. Additionally, the group owner, manager, or your Salesforce system administrator can invite people to join the group using the [CollaborationInvitation](#) object.

The Salesforce system administrator doesn't need to be a member of the group in order to send invitations using the API.

SEE ALSO:

[CollaborationGroupFeed](#)
[CollaborationGroupMember](#)
[CollaborationGroupMemberRequest](#)

CollaborationGroupFeed

Represents a single feed item on a Chatter group feed. A group feed shows posts and comments about the group. This object is available in API version 19.0 and later.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

- “Manage Unlisted Groups”

Only users with this permission can delete items in unlisted groups.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of CollaborationGroupFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>

Field	Details
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>

Field	Details
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDate DESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>

Field	Details
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.

Field	Details
	<ul style="list-style-type: none"> • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the group that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API.</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also

Field	Details
	<p>occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).</p> <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none">• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RipplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• AttachArticleEvent—generated event when a user attaches an article to a case.• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• CaseCommentPost—generated event when a user adds a case comment for a case object.• ChangeStatusPost—generated event when a user changes the status of a case.• ChatTranscriptPost—generated event when Live Agent transcript is saved to a case.• EmailMessageEvent—generated event when an email related to a case object is sent or received.

Field	Details
	<ul style="list-style-type: none">• <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated.• <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status.• <code>SocialPost</code>—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a group.

SEE ALSO:

[CollaborationGroup](#)

[CollaborationGroupMember](#)

[NewsFeed](#)

CollaborationGroupMember

Represents a member of a Chatter group. This object is available in API version 19.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `describeLayout()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
CollaborationGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the associated CollaborationGroup.</p>
CollaborationRole	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The role of a group member. Group owners and managers can change roles for members of their groups. The valid values are:</p> <ul style="list-style-type: none"> • Standard—Indicates that a user is a group member. Members can post and comment in the group. • Admin—Indicates that a user is a group manager. Managers can post and comment, change member roles, edit group settings, add and remove members, delete posts and comments, and edit the group information field. <p> Note: To change the group owner, use the <code>OwnerId</code> field on the CollaborationGroup object.</p>
LastFeedAccessDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when a group member last accessed the group's feed. The value is only updated when a member explicitly consumes the group's feed, not when the member sees group posts in other feeds, like the profile feed.</p>
MemberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field	Details
	<p>Description ID of the group member.</p>
NotificationFrequency	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Required. The frequency at which Salesforce sends Chatter group email digests to this member. Can only be set by the member or users with the "Modify All Data" permission. The valid values are:</p> <ul style="list-style-type: none"> • D—Daily • W—Weekly • N—Never • P—On every post <p>The default value is specified by the member in their Chatter email settings. In communities, the <code>Email on every post</code> option is disabled once more than 10,000 members choose this setting for the group. All members who had this option selected are automatically switched to <code>Daily</code> digests.</p>

Usage

Use this object to view, create, and delete Chatter group members. You must be a group owner or manager to create members for private Chatter groups.

SEE ALSO:

[CollaborationGroup](#)
[CollaborationGroupFeed](#)
[CollaborationGroupMemberRequest](#)

CollaborationGroupMemberRequest

Represents a request to join a private Chatter group. This object is available in API version 21.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CollaborationGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the private Chatter group.</p>
RequesterId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user requesting to join the group; must be the ID of the context user.</p>
ResponseMessage	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Optional message to be included in the notification email when <code>Status</code> is Declined.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status of the request. Available values are:</p> <ul style="list-style-type: none"> • Accepted • Declined • Pending

Usage

This object represents a request to join a private Chatter group, and can be used to accept or decline requests to join private groups you own or manage. On create, an email is sent to the owner and managers of the private group to be accepted or declined. When the `Status` is Accepted or Declined, an email is sent to notify the requester. When the `Status` is Declined, a `ResponseMessage` is optionally included to provide additional details.

Note the following when working with requests:

- Users with the “Modify All Data” or “View All Data” permission can view records for all groups, regardless of membership.
- A user can be a member of 300 groups. Requests to join groups count against this limit.
- `status` can't be specified on create.
- You can only update a request when the `status` is `Pending`.
- You can't delete or update a request with a `status` of `Accepted` or `Declined`.

SEE ALSO:

[CollaborationGroup](#)[CollaborationGroupMember](#)

CollaborationGroupRecord

Represents the records associated with Chatter groups.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), query(), retrieve(), update(), upsert(),
undelete()
```

Fields

Field	Details
<code>CollaborationGroupId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Chatter group.</p>
<code>NetworkId</code>	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Optional. The ID of the community that the group belongs to. Available from API version 34.0.</p>
<code>RecordId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field	Details
	<p>Description Required. The ID of the record associated with the Chatter group.</p>

CollaborationInvitation

Represents an invitation to join Chatter, either directly or through a group. This object is available in API version 21.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Invitations are available if “Allow Invitations” is enabled for your organization.

Invitations are limited to your allowed domain(s) unless the invite is sent from a private group that allows customers. Allowed domains are set by the administrator.

Invitations to customers are available if “Allow Customer Invitations” is enabled for your organization. Users must have the “Invite Customers to Chatter” permission to send invitations to people outside their Chatter domain.

Fields

Field	Details
InvitedUserEmail	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The email address for the user invited to join Chatter. Label is <code>Invited Email</code>.</p>
InvitedUserEmailNormalized	<p>Type email</p> <p>Properties Filter, Group, Sort</p> <p>Description A normalized version of the <code>InvitedUserEmail</code> entered. Label is <code>Invited Email (Normalized)</code>.</p>
InviterId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The person that initiated the invitation.</p>
OptionalMessage	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description An optional message from the person sending the invitation to the person receiving it.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Used when the email address on the invitation is different than the one entered when the invitee accepts the invitation.</p>
SharedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group associated with this invitation.</p> <ul style="list-style-type: none"> • If the invitation is to join Chatter, the <code>SharedEntityId</code> is the ID of the User that created the invitation. The invitee will auto-follow the inviter. • If the invitation is to join a group within Chatter, the <code>SharedEntityId</code> is the ID of the Chatter CollaborationGroup. • To invite a customer, set <code>SharedEntityId</code> to the ID of the private CollaborationGroup with Allow Customers turned on.
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The status of the invitation. Possible values are:</p> <ul style="list-style-type: none"> • Sent • Accepted

Field	Details
	<ul style="list-style-type: none"> • Canceled

Usage

Use this object to create or delete (cancel) invitations to join Chatter. You can either invite a user to join Chatter directly or as part of a CollaborationGroup.

 **Note:** To invite someone to join a CollaborationGroup, you must be either the owner or a manager of the group or a Salesforce system administrator.

The Salesforce system administrator doesn't need to be a member of the group in order to send invitations using the API.

When the person accepts your CollaborationGroup invitation, they join the CollaborationGroup and Chatter as well.

 **Note:** You can't send invitations to users of the organization the invite was sent from.

Invited users can view profiles, post on their feed, and join groups, but can't see your Salesforce data or records.

If your organization allows groups with customers, owners and managers of private groups with the "Allow Customers" setting, as well as system administrators, can use this object to invite customers.

Java Samples

The following example shows how to send an invitation to join Chatter:

```
public void invitePeople(String inviterUserId, String invitedEmail) throws Exception {
    CollaborationInvitation invitation = new CollaborationInvitation();
    invitation.setSharedEntityId(inviterUserId); //pass the userId of the inviter
    invitation.setInvitedUserEmail(invitedEmail); //email of the invited user
    insert(invitation);
}
```

The following example shows how to send an invitation to a customer user from a group that allows customers:

```
public void inviteToGroup(String GroupName, String invitedEmail) throws Exception {
    QueryResult qr = query("select id from collaborationgroup where name = '" +
        GroupName); //pass the group name
    String groupId = qr.getRecords()[0].getId();
    CollaborationInvitation invitation = new CollaborationInvitation();
    invitation.setSharedEntityId(groupId); //pass the groupId
    invitation.setInvitedUserEmail(invitedEmail); //email of the invited user
    insert(invitation);
}
```

Apex Samples

```
String emailAddress = 'bob@external.com';
CollaborationGroup chatterGroup = [SELECT Id
    FROM CollaborationGroup
    WHERE Name='All acme.com'
    LIMIT 1];
```

```

CollaborationInvitation inv = New CollaborationInvitation();
inv.SharedEntityId = chatterGroup.id;
inv.InvitedUserEmail = emailAddress;

try {
    Insert inv;
} catch(DMLException e) {
    System.debug('There was an error with the invite: '+e);
}

```

CombinedAttachment

This read-only object contains all notes, attachments, Google Docs, documents uploaded to libraries in Salesforce CRM Content, and files added to Chatter that are associated with a record.

Supported Calls

`describeSObjects()`

Fields

Field Name	Details
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the document in bytes.</p>
ContentUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL for links and Google Docs. This field is set only for links and Google Docs, and is one of the fields that determine the <code>FileType</code>. This field is available in API version 31.0 and later.</p>
ExternalDataSourceName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	Name of the external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.
	This field is available in API version 32.0 and later.
ExternalDataSourceType	Type
	picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	Type of external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.
	This field is available in API version 35.0 and later.
FileExtension	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	File extension of the document.
	This field is available in API version 31.0 and later.
FileType	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Type of document, determined by the file extension.
ParentId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	The ID of the parent object.
RecordType	Type
	string
	Properties
	Filter, Group, Nillable, Sort

Field Name	Details
	Description
SharingOption	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Controls whether or not sharing is frozen for a file. Only administrators and file owners with Collaborator access to the file can modify this field. Default is Allowed, which means that new shares are allowed. When set to Restricted, new shares are prevented without affecting existing shares.</p> <p>This field is available in API versions 35.0 and later.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Title of the attached file.</p>

Usage

Use this object to list all notes, attachments, documents uploaded to libraries in Salesforce CRM Content, and files added to Chatter for a record, such as a related list on a detail page.

To determine if an object supports the CombinedAttachment object, call `describeSObject()` on the object. For example, `describeSObject('Account')` returns all the child relationships of the Account object, including `CombinedAttachment`. You can then query the `CombinedAttachment` child relationship.

```
SELECT Name, (SELECT Title FROM CombinedAttachments)
FROM Account
```

You can't directly query `CombinedAttachment`.

Community (Zone)

Represents a zone that contains Idea or Question objects.

 **Note:** Starting with the Summer '13 release, Chatter Answers and Ideas "communities" have been renamed to "zones." In API version 28, the API object label has changed to `Zone`, but the API type is still `Community`.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
CanCreateCase	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether users can ask private questions in the zone using Chatter Answers.</p>
DataCategoryName	<p>Type string</p> <p>Properties Filter, Nullable, Group, Sort</p> <p>Description The data category associated with the zone.</p>
Description	<p>Type textarea</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Text description of the zone.</p>
HasChatterService	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether Chatter Answers is available in the zone.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the zone is active or inactive. An idea or question can only be posted to an active zone.</p>

Field	Details
IsPublished	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the zone is available in portals.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The name of the zone.</p>
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the Chatter community that this zone is associated with. This field is available only if Salesforce Communities is enabled in your org. This field is available in API version 38.0 and later.</p>

Usage

Use this object to create a zone in Ideas, Chatter Answers, or Answers. Zones help organize ideas and questions into logical groups and are shared by the Ideas, Answers, and Chatter Answers.

ConnectedApplication

Represents a connected app and its details; all fields are read-only.

Connected apps link client applications, third-party services, other Salesforce organizations, apps and resources to your organization. The connected app configuration specifies authorization and security settings for these resources. This object exposes the settings for a specified connected app.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
MobileSessionTimeout	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Length of time after which the system logs out inactive mobile users.</p>
MobileStartUrl	<p>Type url</p> <p>Properties Filter, idLookup, Nillable, Sort</p> <p>Description Users are directed to this URL after they've authenticated when the app is accessed from a mobile device.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name for this object.</p>
OptionsAllowAdminApprovedUsersOnly	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether access is limited to users granted approval to use the connected app by an administrator. Manage profiles for the app by editing each profile's Access list.</p>
OptionsHasSessionLevelPolicy	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Specifies whether the connected app requires a High Assurance level session.</p>

Field Name	Details
OptionsRefreshTokenValidityMetric	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Specifies whether the refresh token validity is based on duration or inactivity. If true, the token validity is measured based on the last use of the token; otherwise, it is based on the token duration.</p>
PinLength	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description For mobile apps, this is the PIN length requirement for users of the connected app. Valid values are 5, 6, 7, 8 or 9.</p>
RefreshTokenValidityPeriod	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The duration of an authorization token until it expires in hours, months or days as set in the connected app management page.</p>
StartUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description If the app is not accessed from a mobile device, users are directed to this URL after they've authenticated.</p>

Contact

Represents a contact, which is an individual associated with an account.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), merge(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

Customer Portal users can access only portal-enabled contacts.

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the account that is the parent of this contact. We recommend that you only update up to 50 contacts simultaneously when changing the accounts on contacts enabled for a Customer Portal or partner portal. We also recommend that you make this update during times outside of your organization's business hours.</p>
AssistantName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The name of the assistant.</p>
AssistantPhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The telephone number of the assistant.</p>
Birthdate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The birthdate of the contact.</p>

Field	Details
	<p>The year portion of the <code>Birthdate</code> field is ignored in filter criteria, including report filters, list view filters, and SOQL queries. For example, the following SOQL query returns contacts with birthdays later in the year than today:</p> <pre data-bbox="584 382 915 470">SELECT Name, Birthdate FROM Contact WHERE Birthdate > TODAY</pre>
CanAllowPortalSelfReg	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether this contact can self-register for your organization's Customer Portal (<code>true</code>) or not (<code>false</code>).</p>
CleanStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the record's clean status as compared with Data.com. Values are: <code>Matched</code>, <code>Different</code>, <code>Acknowledged</code>, <code>NotFound</code>, <code>Inactive</code>, <code>Pending</code>, <code>SelectMatch</code>, or <code>Skipped</code>. Several values for <code>CleanStatus</code> display with different labels on the contact record detail page.<ul data-bbox="572 1241 1041 1353" style="list-style-type: none">• <code>Matched</code> displays as <code>In Sync</code>• <code>Acknowledged</code> displays as <code>Reviewed</code>• <code>Pending</code> displays as <code>Not Compared</code></p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nillable</p>

Field	Details
	<p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
Department	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The department of the contact.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description A description of the contact. Label is Contact Description. Limit: 32 KB.</p>
DoNotCall	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates that the contact does not wish to be called.</p>
Email	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Email address for the contact.</p>
EmailBouncedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description If bounce management is activated and an email sent to the contact bounces, the date and time the bounce occurred.</p>

Field	Details
EmailBouncedReason	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description If bounce management is activated and an email sent to the contact bounces, the reason the bounce occurred.</p>
Fax	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Fax number for the contact. Label is Business Phone.</p>
FirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description First name of the contact. Maximum size is 40 characters.</p>
HasOptedOutOfEmail	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the contact would prefer not to receive email from Salesforce (<code>true</code>) or not (<code>false</code>). Label is Email Opt Out.</p>
HasOptedOutOfFax	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates that the contact does not wish to receive faxes.</p>
HomePhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description Home telephone number for the contact.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsEmailBounced	<p>Type boolean</p> <p>Properties Filter, Group, Sort</p> <p>Description If bounce management is activated and an email is sent to a contact, indicates whether the email bounced (<code>true</code>) or not (<code>false</code>).</p>
IsPersonAccount	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Read only. Label is Is Person Account. Indicates whether this account has a record type of Person Account (<code>true</code>) or not (<code>false</code>).</p>
Jigsaw	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description References the ID of a company in Data.com. If an account has a value in this field, it means that the account was imported from Data.com. If the field value is <code>null</code>, the account was not imported from Data.com. Maximum size is 20 characters. Available in API version 22.0 and later. Label is Data.com Key.</p> <p> Important: The Jigsaw field is exposed in the API to support troubleshooting for import errors and reimporting of corrected data. Do not modify this value.</p>
LastActivityDate	<p>Type date</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Value is one of the following, whichever is the most recent:</p> <ul style="list-style-type: none"> • Due date of the most recent event logged against the record. • Due date of the most recently closed task associated with the record.
LastCUREquestDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The last date that a Stay-in-Touch request was sent to the contact.</p>
LastCUUpdateDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The last time a Stay-in-Touch update was processed for the contact.</p>
LastName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Last name of the contact. Maximum size is 80 characters.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p>

Field	Details
LeadSource	<p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
MailingAddress	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The source of the lead.</p>
MailingCity	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the mailing address. Read-only. See Address Compound Fields for details on compound address fields.</p>
<ul style="list-style-type: none"> • MailingCity • MailingState • MailingCountry • MailingPostalCode 	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the mailing address.</p>
<ul style="list-style-type: none"> • MailingStateCode • MailingCountryCode 	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO codes for the mailing address's state and country.</p>
MailingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street address for mailing address.</p>
MailingGeocodeAccuracy	<p>Type picklist</p>

Field	Details
	<p>Properties Retrieve, Query, Restricted picklist, Nillable</p> <p>Description Accuracy level of the geocode for the mailing address. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
MailingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>MailingLongitude</code> to specify the precise geolocation of a mailing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
MailingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>MailingLatitude</code> to specify the precise geolocation of a mailing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
MasterRecordId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If this object was deleted as the result of a merge, this field contains the ID of the record that was kept. If this object was deleted for any other reason, or has not been deleted, the value is <code>null</code>.</p>
MiddleName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Middle name of the contact. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>

Field	Details
MobilePhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contact's mobile phone number.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Concatenation of FirstName, MiddleName, LastName, and Suffix. Maximum size is 121 characters.</p>
OtherAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the other address. Read-only. See Address Compound Fields for details on compound address fields.</p>
<ul style="list-style-type: none"> • OtherCity • OtherCountry • OtherPostalCode • OtherState 	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for alternate address.</p>
<ul style="list-style-type: none"> • OtherCountryCode • OtherStateCode 	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO codes for the alternate address's state and country.</p>
OtherGeocodeAccuracy	<p>Type picklist</p> <p>Properties Retrieve, Query, Restricted picklist, Nillable</p>

Field	Details
	<p>Description Accuracy level of the geocode for the other address. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
OtherLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>OtherLongitude</code> to specify the precise geolocation of an alternate address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
OtherLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>OtherLatitude</code> to specify the precise geolocation of an alternate address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
OtherPhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Telephone for alternate address.</p>
OtherStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street for alternate address.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The ID of the owner of the account associated with this contact.</p>

Field	Details
Phone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Telephone number for the contact. Label is Business Phone.</p>
PhotoUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Path to be combined with the URL of a Salesforce instance (for example, https://yourInstance.salesforce.com/) to generate a URL to request the social network profile image associated with the contact. Generated URL returns an HTTP redirect (code 302) to the social network profile image for the contact. Blank if Social Accounts and Contacts isn't enabled for the organization or if Social Accounts and Contacts is disabled for the requesting user.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
ReportsToId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description This field is not visible if <code>IsPersonAccount</code> is <code>true</code>.</p>
Salutation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Honorous abbreviation, word, or phrase to be used in front of name in greetings, such as Dr. or Mrs.</p>

Field	Details
Suffix	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name suffix of the contact. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Title of the contact such as CEO or Vice President.</p>



Note: If you are importing Contact data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Usage

Use this object to manage individuals who are associated with an Account in your organization. You can create, query, delete, or update any Attachment associated with a contact.

Create or update contacts by converting a Lead with the `convertLead()` call.

SEE ALSO:

[Object Basics](#)
[ContactHistory](#)

ContactCleanInfo

Stores the metadata Data.com Clean uses to determine a contact record's clean status. Helps you automate the cleaning or related processing of contact records. ContactCleanInfo includes a number of bit vector fields.

Contact Clean Info provides a snapshot of the data in your Salesforce contact record and its matched Data.com record at the time the Salesforce record was cleaned.

Contact Clean Info includes a number of bit vector fields, whose component fields each correspond to individual object fields and provide related data or status information about those fields. For example, the bit vector field `IsDifferent` has an `IsDifferentEmail` field. If the `IsDifferentEmail` field's value is `False`, that means the `Email` field value is *the same* on the Salesforce contact record and its matched Data.com record.

ContactCleanInfo bit vector fields include:

- `CleanedBy` indicates who (a user) or what (a Clean job) cleaned the contact record.

- `IsDifferent` indicates whether or not a field on the contact record has a value that differs from the corresponding field on the matched Data.com record.
- `IsFlaggedWrong` indicates whether or not a field on the contact record has a value that is flagged as wrong to Data.com.
- `IsReviewed` indicates whether or not a field on the contact record is in a `Reviewed` state, which means that the value was reviewed but not accepted.

Fields

Field Name	Details
Address	<p>Type address</p> <p>Properties Filter, Nullable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
City	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Details for the billing address of the contact.</p>
CleanedByJob	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the contact record was cleaned by a Data.com Clean job (<code>true</code>) or not (<code>false</code>).</p>
CleanedByUser	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the contact record was cleaned by a Salesforce user (<code>true</code>) or not (<code>false</code>).</p>
ContactId	<p>Type reference</p>

Field Name	Details
ContactStatusDataDotCom	<p>Properties Filter, Group, Sort</p> <p>Description The unique, system-generated ID assigned when the contact record was created.</p>
Country	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The status of the contact per Data.com. Values are: Contact is Active per Data.com, Phone is Wrong per Data.com , Email is Wrong per Data.com, Phone and Email are Wrong per Data.com, Contact Not at Company per Data.com, Contact is Inactive per Data.com, Company this contact belongs to is out of business per Data.com, Company this contact belongs to never existed per Data.com or Email address is invalid per Data.com.</p>
DataDotComID	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address of the contact.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID Data.com maintains for the contact.</p>
FirstName	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The contact's first name.</p>
IsDifferentCity	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description Indicates whether the contact's <code>City</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountry	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description Indicates whether the contact's <code>Country</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountryCode	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description Indicates whether the contact's <code>Country Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentEmail	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description Indicates whether the contact's <code>Email</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentFirstName	<p>Type boolean</p>
	<p>Properties Filter</p>

Field Name	Details
IsDifferentLastName	Description
	Indicates whether the contact's <code>First Name</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentPhone	Type
	boolean
Properties	
	Filter
Description	
	Indicates whether the contact's <code>Last Name</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentPostalCode	Type
	boolean
Properties	
	Filter
Description	
	Indicates whether the contact's <code>Postal Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentState	Type
	boolean
Properties	
	Filter
Description	
	Indicates whether the contact's <code>State</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentStateCode	Type
	boolean
Properties	
	Filter

Field Name	Details
	<p>Description Indicates whether the contact's <code>State_Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
<code>IsDifferentStreet</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the contact's <code>Street</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
<code>IsDifferentTitle</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the contact's <code>Title</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongAddress</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the contact's <code>Address</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongEmail</code>	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the contact's <code>Email</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongName</code>	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	<p>Description Indicates whether the contact's <code>Name</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongPhone</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the contact's <code>Phone</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongTitle</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the contact's <code>Title</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsInactive</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the contact has been reported to Data.com as <code>Inactive</code> (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedAddress</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the contact's <code>Address</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedEmail</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p>

Field Name	Details
	<p>Description Indicates whether the contact's Email field value is in a Reviewed state (true) or not (false).</p>
IsReviewedName	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the contact's Name field value is in a Reviewed state (true) or not (false).</p>
IsReviewedPhone	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the contact's Phone field value is in a Reviewed state (true) or not (false).</p>
IsReviewedTitle	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the contact's Title field value is in a Reviewed state (true) or not (false).</p>
LastMatchedDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date the contact record was last matched and linked to a Data.com record.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The contact's last name.</p>

Field Name	Details
LastStatusChangedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of who or what last changed the record's Clean Status field value: a Salesforce user or a Clean job.</p>
LastStatusChangedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date on which the record's Clean Status field value was last changed.</p>
Latitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with Longitude to specify the precise geolocation of a billing address. Data not currently provided.</p>
Longitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with Latitude to specify the precise geolocation of a billing address. Data not currently provided.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description Field label is Contact Clean Info Name. The name of the contact. Maximum size is 255 characters.</p>
Phone	<p>Type phone</p>

Field Name	Details
	Properties
	Filter, Group, Nillable, Sort
	Description
	The phone number for the contact.
PostalCode	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the contact.
State	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the contact.
Street	Type
	textarea
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the contact.
Title	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The contact's title.

Usage

Developers can create triggers that read the Contact Clean Info fields to help automate the cleaning or related processing of contact records.

Create a customized set of `Title` field values. Use triggers to map values from fields on imported or cleaned records onto a standard set of values.

ContactFeed

Represents a single feed item in the feed on a contact record detail page. This object is available in API version 18.0 and later.

A contact feed shows recent changes to a contact record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to Contact records.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Contact object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of ContactFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p>

Field	Details
	<p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p>

Field	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.

Field	Details
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the contact record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used.

Field	Details
	<ul style="list-style-type: none"> • <code>CanvasPost</code>—a post made by a canvas app posted on a feed. • <code>CollaborationGroupCreated</code>—generated when a user creates a public group. • <code>CollaborationGroupUnarchived</code>—Not used. • <code>ContentPost</code>—a post with an attached file. • <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher. • <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold. • <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed. • <code>LinkPost</code>—a post with an attached URL. • <code>PollPost</code>—a poll posted on a feed. • <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile. • <code>QuestionPost</code>—generated when a user posts a question. • <code>ReplyPost</code>—generated when Chatter Answers posts a reply. • <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated.
	<p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case
	 Note: If you set <code>Type</code> to <code>ContentPost</code> , also specify <code>ContentData</code> and <code>ContentFileName</code> .
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p>

Field	Details
	<ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a contact record.

SEE ALSO:

[Contact](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

ContactHistory

Represents the history of changes to the values in the fields of a contact. This object is available in versions 11.0 and later.

Supported Calls

`describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
ContactId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Contact. Label is Contact ID.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to a contact.

This object respects field level security on the parent object.

SEE ALSO:

[Contact](#)

ContactOwnerSharingRule

Represents the rules for sharing a contact with a User other than the owner.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
ContactAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group, UserRole, or User for Contacts. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. A Contact owned by a User in the source Group triggers the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the User or Group being granted access.</p>

Usage

Use this object to manage the sharing rules for contacts.

SEE ALSO:

- [Contact](#)
- [ContactShare](#)
- [Metadata API Developer Guide: SharingRules](#)

ContactShare

Represents a list of access levels to a Contact along with an explanation of the access level. For example, if you have access to a record because you own it, the `ContactAccessLevel` is `All` and `RowCause` is `Owner`.

Supported Calls

`describeSObjects(), create(), delete(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ContactId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Contact associated with this sharing entry. This field can't be updated.</p>
ContactAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to cases associated with the account Contact. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All This value is not valid for create or update. <p>This field must be set to an access level that is higher than the organization's default access level for contacts.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). There are many possible values, including:</p> <ul style="list-style-type: none"> • <code>Rule</code>—The User or Group has access via a Contact sharing rule. • <code>ImplicitChild</code>—The User or Group has access to the Contact via sharing access on the associated Account. • <code>Manual</code>—The User or Group has access because a User with “All” access manually shared the Contact with them. • <code>Owner</code>—The User is the owner of the Contact.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description</p> <p>ID of the User or Group that has been given access to the Contact. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit Contact records owned by other users.

SEE ALSO:

[AccountShare](#)

ContactTag

Associates a word or short phrase with a Contact.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p>

Field Name	Details
	<p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

ContactTag stores the relationship between its parent TagDefinition and the Contact being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

ContentAsset

Represents a Salesforce file that has been converted to an asset file in a custom app inLightning Experience. Enables a Salesforce file to be used for org setup and configuration purposes. Asset files can be packaged and referenced by other components. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`

Special Access Rules

- Only admin users can create, edit, or delete ContentAssets.
- It isn't necessary to create asset files for regular, collaborative use of Salesforce Files. "Assetize" files only when they're used in setup and configuration situations.
- Neither the file (ContentDocument) nor the asset settings record (ContentAssets) can be deleted if the asset file is referenced by another component.
- ContentAsset doesn't support search or most recently used (MRU) lists.
- ContentAsset doesn't support Apex triggers.
-

Fields

Field	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the document.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the asset file in the API. ContentAsset.DeveloperName:</p> <ul style="list-style-type: none">must be 40 characters or fewermust begin with a lettercan contain only underscores and alphanumeric characterscan't include spacescan't end with an underscore

Field	Details
	<ul style="list-style-type: none"> can't contain 2 consecutive underscores <p>In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language for this document. This field defaults to the user's language unless the org is multi-language enabled. Specifies the language of the labels returned. The value must be a valid user locale (language and country), such as "de_DE" or "en_GB". For more information on locales, see the <code>Language</code> field on the <code>CategoryNodeLocalization</code> object.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The master label for the asset file. This internal label doesn't get translated.</p>
NameSpacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p>

ContentDistribution

Represents information about sharing a document externally. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Salesforce CRM Content must be enabled to query shared documents. To query content deliveries, content deliveries must be enabled.
- Users (including users with the “View All Data” permission) can query only the files that they have access to. If the file is managed by a Content Library, the user must have “Deliver Content” enabled in the library permission definition and be a member of the library. If the file isn’t managed by a Content Library, the user must have the “Enable Creation of Content Deliveries for Salesforce Files” permission.
- Users can query the `DistributionPublicUrl` and `Password` fields only if they are the file owner, if the file is shared with them, or if the `RelatedRecordId` specifies a record that the users can access.
- If the shared document is deleted, the delete cascades to any associated ContentDistribution. The ContentDistribution is still queryable by using the `QueryAll` verb.
- If the shared document is archived, the only fields that users can edit are `ExpiryDate` and `PreferencesExpires`.
- Customer Portal users can’t access this object.
- Chatter Free users can’t access this object.

Fields

Field Name	Details
<code>ContentDocumentId</code>	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the shared document.</p>
<code>ContentVersionId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the shared document version.</p>
<code>DistributionPublicUrl</code>	<p>Type string</p> <p>Properties Nillable, Sort</p> <p>Description URL of the link to the shared document.</p>
<code>ExpiryDate</code>	<p>Type dateTime</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Date when the shared document becomes inaccessible.</p>
FirstViewDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date when the shared document is first viewed.</p>
LastViewDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date when the shared document was last viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the content delivery.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns the shared document.</p>
Password	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description A password that allows access to a shared document.</p>
PreferencesAllowOriginalDownload	<p>Type boolean</p>

Field Name	Details
	Properties Create, Filter, Update
	Description When <code>true</code> , the shared document can be downloaded as the file type that it was uploaded as. When <code>false</code> , download availability depends on whether a preview of the file exists. If a preview exists, the file can't be downloaded. If a preview doesn't exist, the file can still be downloaded. If the shared document is a link, it can't be downloaded.
PreferencesAllowPDFDownload	Type boolean
	Properties Create, Filter, Update
	Description When <code>true</code> , the shared document can be downloaded as a PDF if the original file type is PDF or if a PDF preview has been generated.
PreferencesAllowViewInBrowser	Type boolean
	Properties Create, Filter, Update
	Description When <code>true</code> , a preview of the shared document can be viewed in a Web browser.
PreferencesExpires	Type boolean
	Properties Create, Filter, Update
	Description When <code>true</code> , access to the shared document expires on the date that's specified by <code>ExpiryDate</code> .
PreferencesLinkLatestVersion	Type boolean
	Properties Create, Filter, Update
	Description When <code>true</code> , users see the most recent version of a shared document. When <code>false</code> , users see the version of the document that's shared, even if it isn't the most recent version.

Field Name	Details
PreferencesNotifyOnVisit	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the owner of the shared document is emailed the first time that someone views or downloads the shared document.</p>
PreferencesNotifyRndtnComplete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the owner of the shared document is emailed when renditions of the shared document that can be previewed in a Web browser are generated.</p>
PreferencesPasswordRequired	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, a password, specified by <code>Password</code>, is required to access the shared document.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the record, such as an Account, Campaign, or Case, that the shared document is related to.</p>
ViewCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of times that the shared document has been viewed.</p>

Usage

Use this object to create, update, delete, or query information about a document shared externally via a link or via Salesforce CRM Content delivery.

The ContentDistribution object supports triggers before and after these operations: insert, update, delete. It supports triggers after undelete.

 **Example:** The VP of Marketing wants file authors to specify whether their files can be shared with external people using content delivery. He also wants some files to have a password. You can add a custom field `DeliveryPolicy` on the ContentVersion object. Make the custom field a picklist with the values, `Allowed`, `Blocked`, and `Password required`. Add the field to the ContentVersion layout so that the user can set the delivery policy per file. Then, add an insert trigger for the ContentDistribution object to enforce the rules based on the delivery policy set in the file.

 **Note:** The ContentVersionId for ContentDistribution must be unique.

This trigger for the ContentDistribution object enforces the delivery policy rules for each file:

```
trigger deliveryPolicy on ContentDistribution (before insert) {
    for (ContentDistribution cd : trigger.new) {
        String versionId = DeliveryPolicyHelper.getContentVersionId(cd);
        ContentVersion version = [select DeliveryPolicy__c from ContentVersion where
Id = :versionId];
        String policy = version.DeliveryPolicy__c;
        if (policy.equals('Blocked')) {
            cd.addError('This file is not allowed to be delivered.');
        } else if (policy.equals('Password required')){
            if (!DeliveryPolicyHelper.requirePassword(cd)) {
                cd.addError('To deliver this file, set a password.');
            }
        }
    }
}
```

The trigger calls this helper class:

```
public class DeliveryPolicyHelper {
    public static String getContentVersionId(ContentDistribution cd) {
        if (cd.ContentVersionId != null) {
            return cd.ContentVersionId;
        } else {
            String versionId = [select LatestPublishedVersionId from ContentDocument
where Id = :cd.ContentDocumentId].get(0).LatestPublishedVersionId;
            return versionId;
        }
    }

    public static boolean requirePassword(ContentDistribution cd) {
        return cd.PreferencesPasswordRequired;
    }
}
```

 **Important:** Apex has a per organization limit of 10 concurrent requests that last longer than 5 seconds. A trigger that uploads files can easily hit this limit.

ContentDistributionView

Represents information about views of a shared document. This read-only object is available in API version 32.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

- Salesforce CRM Content must be enabled to query shared documents. To query content deliveries, content deliveries must be enabled.
- Users (including users with the “View All Data” permission) can query only the files that they have access to. If the file is managed by a Content Library, the user must have “Deliver Content” enabled in the library permission definition and be a member of the library. If the file isn’t managed by a Content Library, the user must have the “Enable Creation of Content Deliveries for Salesforce Files” permission.
- If the shared document is deleted, the delete cascades to any associated ContentDistributionView. The ContentDistributionView is still queryable by using the `QueryAll` verb.
- Customer Portal users can’t access this object.
- Chatter Free users can’t access this object.

Fields

Field Name	Details
DistributionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the content delivery that the document is part of.</p>
IsDownload	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> if the shared document is downloaded; <code>false</code> if the shared document is viewed.</p>
IsInternal	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description true if the shared document is viewed by a user in the same organization; false if viewed by an external user.</p>
ParentViewId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of this instance of accessing the shared document.</p>

Usage

Use this read-only object to query information about users who are accessing shared documents.

ContentDocument

Represents a document that has been uploaded to a library in Salesforce CRM Content or Salesforce Files. This object is available in API versions 17.0 and later for Salesforce CRM Content. This object is available in API version 21.0 and later for Salesforce Files.

The maximum number of documents that can be published is 10,000,000. Archived files count towards this limit and towards storage usage limits.

- Contact Manager, Group, Professional, Enterprise, Unlimited, and Performance Edition customers can publish a maximum of 36,000 new versions per 24-hour period.
- Developer Edition and trial users can publish a maximum of 2,500 new versions per 24-hour period.

Supported Calls

`delete(), describeLayout() describeSObjects(), query(), retrieve(), search(), undelete(), update()`

Special Access Rules

- Customer and Partner Portal users must have the “View Content in Portal” permission in order to query content in libraries where they have access.
- Users (including users with the “View All Data” permission) can only query files they have access to, including:
 - All Salesforce CRM Content files in libraries they’re a member of and in their personal library, regardless of library permissions (API version 17.0 and later).
 - All Salesforce Files they own, posted on their profile, posted on groups they can see, and shared directly with them (API version 21.0 and later).
- A Salesforce CRM Content document can be deleted if any of the following are true:
 - The document is published into a personal library or is in the user’s upload queue.

- The document is published into a public library, the user is a member of that library with the “Add Content” library privilege enabled, and the user trying to delete the document is the owner.
- The document is published into a public library that has the “Delete Content” or “Manage Library” permission enabled, and the user trying to delete the document is not the owner.

For API version 25.0 and later, you can change ownership of Salesforce Files and Salesforce CRM Content documents.

- The following must be true to change ownership of a Salesforce CRM Content document:
 - The Salesforce CRM Content app must be enabled.
 - The user who is becoming the owner of the document must have a Salesforce CRM Content feature license.
- A user can change ownership of a Salesforce CRM Content document or Salesforce file if any of the following are true:
 - The user is the current owner, or has either the “Modify All Data” or “Manage Salesforce CRM Content” permission enabled.
 - The user has the “Manage Library” permission enabled for the library containing the document.



Note:

- The user who is becoming the owner of the document must be a visible user who is active, but the original owner can be inactive.
- A document's owner can be changed to a user who doesn't have access to the library that contains the document. Library administrators may need to give the new owner membership to the library.

Fields

Field	Details
ArchivedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the document. This field is available in API version 24.0 and later.</p>
ArchivedDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The date when the document was archived. This field is available in API version 24.0 and later.</p>
ContentAssetId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description If the ContentDocument is an asset file, this field points to the asset. For most entities, the value of this field is <code>null</code>. This field is available in API version 38.0 and later.</p>
ContentModifiedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date the document was modified. ContentModifiedDate updates when, for example, the document is renamed or a new document version is uploaded. When you're uploading the first version of a document, ContentModifiedDate can be set to the current time or any time in the past. This field is available in API version 32.0 and later.</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The size of the document in bytes. This field is available in API version 31.0 and later.</p>
Description	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort</p> <p>Description A description of the document. This field is available in API version 31.0 and later.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p>

Field	Details
Description	
	A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.
FileExtension	Type string
Properties	
	Filter, Group, Nillable, Sort
Description	
	File extension of the document.
	This field is available in API version 31.0 and later.
FileType	Type string
Properties	
	Filter, Group, Nillable, Sort
Description	
	Type of document, determined by the file extension.
	This field is available in API version 31.0 and later.
IsArchived	Type boolean
Properties	
	Defaulted on create, Filter, Group, Sort, Update
Description	
	Indicates whether the document has been archived (<code>true</code>) or not (<code>false</code>).
LastReferencedDate	Type date
Properties	
	Filter, Nillable, Sort, Update
Description	
	The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date
Properties	
	Filter, Nillable, Sort, Update

Field	Details
	<p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
LatestPublishedVersionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the latest document version (ContentVersion).</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the owner of this document.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the library that owns the document. Created automatically when inserting a ContentVersion via the API for the first time. This field is available in API version 24.0 and later when Salesforce CRM Content is enabled.</p>
PublishStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Indicates if and how the document is published. Valid values are:</p> <ul style="list-style-type: none"> • P—The document is published to a public library and is visible to other users. Label is Public. • R—The document is published to a personal library and is not visible to other users. Label is Personal Library. • U—The document is not published because publishing was interrupted. Label is Upload Interrupted.

Field	Details
SharingOption	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Controls whether or not sharing is frozen for a file. Only administrators and file owners with Collaborator access to the file can modify this field. Default is <code>Allowed</code>, which means that new shares are allowed. When set to <code>Restricted</code>, new shares are prevented without affecting existing shares.</p> <p>This field is available in API versions 35.0 and later.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The title of a document.</p>

Usage

- Use this object to retrieve, query, update, and delete the latest version of a document, but not a content pack, in a library or a Salesforce file. Use the ContentVersion object to create, query, retrieve, search, edit, and update a specific version of a Salesforce CRM Content document or Salesforce file.
- The `query()` call doesn't return archived documents. The `queryAll()` call returns archived documents.
- You can't add new versions of archived documents.
- To query a file that is shared only with a record, you must specify the content ID of the file.
- To create a document, create a new version via the ContentVersion object without setting the `ContentDocumentId`. This automatically creates a parent document record. When adding a new version of the document, you must specify an existing `ContentDocumentId` which initiates the revision process for the document. When the latest version is published, the title, owner, and publish status fields are updated in the document.
- When you delete a document, all versions of that document are deleted, including ratings, comments, and tags.
- You can't create, edit, or delete content packs via the API.



Note: Content metadata, such as tags, custom fields, and content owners are tracked at the version level rather than at the document level.

- If you query versions in the API, versions with a `PublishStatus` of `Upload Interrupted` are not returned.
- A document record is a container for multiple version records. You create a new version to add a document to the system. The new version contains the actual file data which allows the document to have multiple versions. The version stores the body of the uploaded document.

- Assign topics to ContentDocument using `TopicAssignment` in API version 37.0 or later.

SEE ALSO:

[ContentDocumentFeed](#)

[ContentDocumentHistory](#)

[ContentVersion](#)

ContentDocumentFeed

Represents a single feed item associated with ContentDocument. A content document feed shows these content document changes: creating a ContentDocument file and uploading a new ContentDocument. This object is available in versions 20.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the ContentDocument object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of ContentDocumentFeed. Required when Type is <code>TextPost</code>. Optional when Type is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type <code>int</code></p> <p>Properties <code>Filter</code>, <code>Group</code>, <code>Sort</code></p>

Field	Details
	<p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
	Description
	Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.
ContentType	Type string
	Properties
	Group , Nillable , Sort
	Description
	Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.
CreatedDate	Type dateTime
	Properties
	Defaulted on create, Filter, Sort
	Description
	Date and time when this record was created. This field is a standard system field.
	Ordering by <code>CreatedDate DESC</code> sorts the feed by the most recent feed item.
FeedPostId	Type reference
	Properties
	Filter , Group , Nillable , Sort
	Description
	This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only.
	ID of the associated FeedPost . A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.
InsertedById	Type reference
	Properties
	Filter , Group , Nillable , Sort
	Description
	ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.
IsDeleted	Type boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (true) or not (false).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item Body contains rich text. Set IsRichText to true if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field.</p>

Field	Details
	<p>When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>.</p> <p>Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Create, Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the document that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. For Work.com thanks posts, it's the ID of the WorkThanks object associated with a RypplePost. This field is typically null for all posts except ContentPost and RypplePost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Title	<p>Type string</p> <p>Properties Create, Group, Nillable, Sort, Update</p> <p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name. The Title field can be updated on posts of Type QuestionPost.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of FeedItem. Except for ContentPost, LinkPost, and TextPost, don't create FeedItem types directly from the API.</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used.

Field	Details
	<ul style="list-style-type: none">• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RipplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none">• AttachArticleEvent—generated event when a user attaches an article to a case.• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• CaseCommentPost—generated event when a user adds a case comment for a case object.• ChangeStatusPost—generated event when a user changes the status of a case.• ChatTranscriptPost—generated event when Live Agent transcript is saved to a case.• EmailMessageEvent—generated event when an email related to a case object is sent or received.• FacebookPost—generated when a Facebook post is created from a case. Deprecated.• MilestoneEvent—generated when a case milestone is completed or reaches violation status.• SocialPost—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Field	Details
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none">• AllUsers—The feed item is available to all users who have permission to see the feed item.• InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none">• For record posts, Visibility is set to InternalUsers for all internal users by default.• External users can set Visibility only to AllUsers.• On user and group posts, only internal users can set Visibility to InternalUsers.

Usage

Use this object to track changes for a ContentDocument object.

SEE ALSO:

[ContentDocument](#)

ContentDocumentHistory

Represents the history of a document. This object is available in versions 17.0 and later.

Supported Calls

`describeObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

- Customer and Partner Portal users must have the “View Content in Portal” permission to query content in libraries where they have access.
- A user can query all versions of a document from their personal library and any version that is part of or shared with a library where they are a member, regardless of library permissions.

Fields

Field	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the document.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Group, Sort, Restricted picklist</p> <p>Description The name of the field that was changed. Possible values include:</p> <ul style="list-style-type: none"> • contentDocPublished—The document is published into a library. • contentDocUnpublished—The document is archived or removed from a library, either directly or when the owning library is changed. • contentDocRepublished—The document is removed from the archive. • contentDocFeatured—The document is featured. • contentDocSubscribed—The document is subscribed to. • contentDocUnsubscribed—The document is no longer subscribed to.
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>

Field	Details
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this read-only object to query the history of a document.

SEE ALSO:

[ContentDocument](#)

ContentDocumentLink

Represents the link between a Salesforce CRM Content document or Salesforce file and where it's shared. A file can be shared with other users, groups, records, and Salesforce CRM Content libraries. This object is available in versions 21.0 and later for Salesforce CRM Content documents and Salesforce Files.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update()`

Special Access Rules

- Customer and Partner Portal users must have the “View Content in Portal” permission to query content in libraries where they have access.
- Users (including users with the “View All Data” permission) can only query files they have access to, including:
 - All Salesforce CRM Content files in libraries they’re a member of and in their personal library, regardless of library permissions (API version 17.0 and later).
 - All Salesforce Files they own, posted on their profile, posted on groups they can see, and shared directly with them (API version 21.0 and later).
- In API versions 33.0 and later, you can create and delete ContentDocumentLink objects with a `LinkedEntityId` of any record type that can be tracked in the feed, even if feed tracking is disabled for that record type.
- In API versions 25.0 and later, you can create ContentDocumentLink objects with a `LinkedEntityId` of type `User`, `CollaborationGroup`, or `Organization`.
- In API versions 21.0 and later, users with explicit Viewer access (the file has been directly shared with the user) to a file can delete ContentDocumentLink objects between the file and other users who have Viewer access. In the same API versions, any user with Viewer access to a file can deleteContentDocumentLink objects between the file and organizations or groups of which they are a member.

- For organizations with Communities enabled, a document can only be shared with users and groups that are a part of the [community](#) the file was created in.

Fields

Field	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the document.</p>
LinkedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the linked object. Can include Chatter users, groups, records (any that support Chatter feed tracking including custom objects), and Salesforce CRM Content libraries.</p>
ShareType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Required. The permission granted to the user of the shared file in a library. This is determined by the permission the user already has in the library. This field is available in API version 25.0 and later.</p> <p>Viewer The user can explicitly view but not edit the shared file.</p> <p>Collaborator The user can explicitly view and edit the shared file.</p> <p>Inferred The user's permission is determined by the related record. For shares with a library, this is defined by the permissions the user has in that library.</p>
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description</p> <p>Specifies whether this file is available to all users, internal users, or shared users. This field is available in API version 26.0 and later.</p> <p>Visibility can have the following values.</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The file is available to all users who have permission to see the file. • <code>InternalUsers</code>—The file is available only to internal users who have permission to see the file. • <code>SharedUsers</code>—The file is available to all users who can see the feed to which the file is posted. <code>SharedUsers</code> is used only for files shared with users, and is available only when an org has private org-wide sharing on by default. The <code>SharedUsers</code> value is available in API version 32.0 and later. <p>Note the following exceptions for Visibility.</p> <ul style="list-style-type: none"> • <code>AllUsers</code> & <code>InternalUsers</code> values apply to files posted on standard and custom object records, but not to users, groups, or content libraries. • For posts to a record feed, <code>visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>. • For posts to a user feed, if the organization-wide default for user sharing is set to private, <code>Visibility</code> is set to <code>SharedUsers</code>. <p>The visibility setting on <code>ContentDocumentLink</code> determines a file's visibility on a record post. When a file has multiple references posted in a feed, the file's visibility is determined by the most visible setting.</p>

Usage

Use this object to query the locations where a file is shared or query which files are linked to a particular location. For example, the following query returns a particular document shared with a Chatter group:

```
SELECT ContentDocument.title FROM ContentDocumentLink WHERE ContentDocumentId = '069D0000000so2' AND LinkedEntityId = '0D500000089123'
```

- You can't run a query without filters against `ContentDocumentLink`.
- You can't filter on `ContentDocument` fields if you're filtering by `ContentDocumentId`. You can only filter on `ContentDocument` fields if you're filtering by `LinkedEntityId`.
- You can't filter on the related object fields. For example, you can't filter on the properties of the account to which a file is linked. You can filter on the properties of the file, such as the `title` field.

A SOQL query must filter on one of `Id`, `ContentDocumentId`, or `LinkedEntityId`.

The `ContentDocumentLink` object supports triggers before and after these operations: insert, update, delete.



Example: This trigger for the ContentDocumentLink object prevents public XLSX files from being shared.

```
trigger NoShareXLSX on ContentDocumentLink (after insert) {
    for (ContentDocumentLink cdl : trigger.new) {
        if (!CDLHelper.isSharingAllowed(cdl)) {
            cdl.addError('Sorry, you cannot share this file.');
        }
    }
}
```

The trigger calls this helper class.

```
public class CDLHelper {

    /**
     * Gets FileExtension of the inserted content.
     */
    public static String getFileExtension(ContentDocumentLink cdl) {
        String fileExtension;
        String docId = cdl.ContentDocumentId;
        FileExtension = [select FileExtension from ContentVersion where ContentDocumentId = :docId].get(0).FileExtension;
        return FileExtension;
    }

    /**
     * Checks the file's PublishStatus and FileExtension to decide whether user can
     * share the file with others.
     * PublishStatus 'P' means the document is in a public library.
     */
    public static boolean isSharingAllowed(ContentDocumentLink cdl) {
        String docId = cdl.ContentDocumentId;
        ContentVersion version = [select PublishStatus,FileExtension from ContentVersion
        where ContentDocumentId = :docId].get(0);
        if (version.PublishStatus.equals('P') && (version.FileExtension != null &&
version.FileExtension.equals('xlsx'))) {
            return false;
        }
        return true;
    }

    /**
     * Gets the parent account name if the file is linked to an account.
     */
    public static String getAccountName(ContentDocumentLink cdl) {
        String name;
        String id = cdl.LinkedEntityId;
        if (id.substring(0,3) == '001') {
            name = [select Name from Account where Id = :id].get(0).Name;
        }
        return name;
    }
}
```

! **Important:** Apex has a per organization limit of 10 concurrent requests that last longer than 5 seconds. A trigger that uploads files can easily hit this limit.

SEE ALSO:

[ContentDocument](#)

ContentFolder

Represents a folder in a content library for adding files. This object is available in API version 34.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Special Access Rules

- Salesforce CRM Content or Chatter must be enabled to access ContentFolder.
- All users with a content feature license can modify folders in their personal library.
- To modify a folder, the user must be a member of the library and have permission to modify folders.

Fields

Field Name	Details
Name	Type string Properties Create, Filter, Group, Sort, Update Description Name of the folder.
ParentContentFolderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the ParentFolder.

ContentFolderItem

Represents a file (ContentDocument) or folder (ContentFolder) that resides in a ContentFolder in a ContentWorkspace. This object is available in API version 35.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Fields

Field Name	Details
ContentSize	Type int
	Properties Filter, Group, Nillable, Sort
	Description The file or folder size of the ContentFolderItem.
FileExtension	Type string
	Properties Filter, Group, Nillable, Sort
	Description Specifies the file extension if the ContentFolderItem is a file.
FileType	Type string
	Properties Filter, Group, Nillable, Sort
	Description Specifies the type of file if ContentFolderItem is a file.
IsFolder	Type boolean
	Properties Defaulted on create, Filter, Group, Sort
	Description Indicates that the ContentFolderItem is a folder, and not a file.
ParentContentFolderId	Type reference
	Properties Filter, Group, Nillable, Sort

Field Name	Details
	Description
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The name of the file or folder.</p>

Usage

ContentFolderLink

Defines the association between a library and its root folder. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

- Salesforce CRM Content must be enabled to access ContentFolderLink.
- ContentFolderLink is read-only in the context of a library.

Fields

Field Name	Details
	Type
ContentFolderId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the folder.</p>
ParentEntityId	<p>Type reference</p>

Field Name	Details
	Properties Filter, Group, Sort
	Description Name of the entity the folder hierarchy is linked to.

ContentFolderMember

Defines the association between a file and a folder. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve(), update()`

Special Access Rules

- Salesforce CRM Content or Chatter must be enabled to access ContentFolderMember.
- All users with a content feature license can modify folders in their personal library.
- To modify ContentFolderMember, the user must be a member of the library and have permission to modify folders.

Fields

Field Name	Details
ChildRecordId	Type reference Properties Filter, Group, Sort Description ID of the file.
ParentContentFolderId	Type reference Properties Filter, Group, Sort, Update Description ID of the folder the file is in.

ContentHubItem

Represents a file or folder in a Files Connect external data source, such as Microsoft SharePoint or OneDrive for Business. This object is available in API version 33.0 and later.

Special Access Rules

Chatter and Files Connect must be enabled for the organization.

Supported Calls

`describeSObjects()`, `query()`, `search()`

Fields

Field Name	Details
ContentHubRepositoryId	<p>Type reference</p> <p>Properties Filter, Group, Nillable</p> <p>Description The ID for the related external data source described by the ContentHubRepository object.</p>
ContentModifiedDate	<p>Type dateTime</p> <p>Properties Nillable</p> <p>Description Date the file or folder content last changed.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable</p> <p>Description File or folder size.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable</p>

Field Name	Details
	Description
	Explanation of item in external data source.
ExternalContentUrl	Type
	url
	Properties
	Group, Nillable
	Description
	The URL of the document content in the external data source.
ExternalDocumentUrl	Type
	url
	Properties
	Group, Nillable
	Description
	The URL of the detail page in the external data source.
ExternalId	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	ID for the file or folder in the external data source.
FileExtension	Type
	string
	Properties
	Group, Nillable
	Description
	File format extension, such as .doc or .pdf
FileType	Type
	string
	Properties
	Group, Nillable
	Description
	Complete file type, such as "Microsoft Word Document."
IsFolder	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort

Field Name	Details
	<p>Description Indicates whether item is a folder or file.</p>
MimeType	<p>Type string</p> <p>Properties Filter, Group, Nillable</p> <p>Description MIME type of the content.</p>
Name	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Name of the file or folder in the external data source.</p>
Owner	<p>Type string</p> <p>Properties Filter, Group, Nillable</p> <p>Description Username of the content owner in the external data source.</p>
ParentId	<p>Type string</p> <p>Properties Filter, Group, Nillable</p> <p>Description The ID of the parent folder for the record. This field isn't returned in queries or searches of the ContentHubItem object. It supports only WHERE clauses, such as the following: <code>WHERE ContentHubRepositoryId = <ID of external source> and ParentId = <ID of parent folder or record>.</code> Or specify <code>WHERE ParentId = <name of root folder></code> to return the children of the root folder.</p> <p> Tip: The ParentId field supports both Salesforce IDs (in the format "0CHxxx") and external IDs.</p>

Field Name	Details
Title	<p>Type string</p> <p>Properties Group, Nullable</p> <p>Description The title that appears in the content, which often differs from the <code>Name</code> of the containing file or folder.</p>
UpdatedBy	<p>Type string</p> <p>Properties Group, Nullable</p> <p>Description Username for the person who last updated the file.</p>

Usage

The following SOQL query examples show how to retrieve files and folders from a Files Connect external data source. These examples use placeholders for ID values for the repository ID and folder IDs. Before running these queries, replace the placeholders with valid ID values for your external data source and folders.

Important: You must filter queries and searches on ContentHubItem with the `ContentHubRepositoryId` field; for example, `SELECT Id FROM ContentHubItem WHERE ContentHubRepositoryId = <ID of external data source>`.

Example 1: Get the ID and name of the root folder in an external file source.

```
SELECT Id, Name
FROM ContentHubItem
WHERE ContentHubRepositoryId = '<repository ID>' AND ParentId = NULL
```

Example 2: List all folders and files under the specified root folder.

```
SELECT Id, Name
FROM ContentHubItem
WHERE ContentHubRepositoryId = '<repository ID>' AND ParentId = '<root folder ID>'
```

Example 3: List all external file data sources by querying ContentHubRepository.

```
SELECT DeveloperName
FROM ContentHubRepository
```

Example 4: List all files and folders in a given folder and external file source.

```
SELECT Id, Name
FROM ContentHubItem
WHERE ContentHubRepositoryId = '<repository ID>' AND ParentId = '<parent folder ID>'
```

Example 5: To return only folders in the result set, add `IsFolder = true` in the WHERE clause to a query that returns files and folders. For example, the following query lists all folders under the root folder.

```
SELECT Id, Name
FROM ContentHubItem
WHERE ContentHubRepositoryId = '<repository ID>' AND ParentId = '<root folder ID>'
AND IsFolder = true
```

Example 6: Retrieve a link that is used to open the specified document in an external source.

```
SELECT ExternalDocumentUrl
FROM ContentHubItem
WHERE ContentHubRepositoryId = '<repository ID>' AND Id = '<document ID>'
```

SOSL Example: Retrieve the ID and name of all documents that contain the search string. The result set is limited to the first 10 documents.

```
FIND {<search string>}
RETURNING ContentHubItem(Id, Name
                         WHERE ContentHubRepositoryId = '<repository ID>')
LIMIT 10
```

ContentHubRepository

Represents a Files Connect external data source such as Microsoft SharePoint or OneDrive for Business. This object is available in API version 33.0 and later.

Special Access Rules

Chatter and Files Connect must be enabled for the organization.

Supported Calls

```
describeLayout(), describeSObjects(), query(), retrieve()
```

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the record in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This field is automatically generated but you can supply your own value if you create the record using the API.</p>

Field Name	Details
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for the external data source. This display value is the internal label and does not get translated.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The data source type. Possible values are:</p> <ul style="list-style-type: none"> • contenthubGoogleDrive • contenthubOffice365 • contenthubOneDrive • contenthubSharepoint

ContentNote

Represents a note in Salesforce. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

- Notes must be enabled.

Fields

Field	Details
Content	<p>Type base64</p> <p>Properties Create, Nullable, Update</p> <p>Description The content or body of the note, which can include properly formatted HTML or plain text. Any special characters within plain text in the Content field must be escaped. You can escape special characters by calling <code>content.escapeHtml4()</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Size of the note in bytes.</p>
CreatedById	<p>Type reference</p> <p>Properties Defaulted on createFilter, Group, Sort</p> <p>Description ID of the user who created the note.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date the note was created.</p>
FileExtension	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description File extension of the note.</p>
FileType	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Type of file for the note. All notes have a file type of SNOTE.</p>
Id	<p>Type id</p> <p>Properties Defaulted on createFilter, Group, Nillable, Sort</p> <p>Description ID of the note.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the note has been deleted.</p>
IsReadOnly	<p>Type boolean</p> <p>Properties Defaulted on create, Group, Sort</p> <p>Description Indicates whether the note is read only.</p>
LastModifiedById	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description The ID of the user who last modified the note.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date the note was last modified. Updates when the Title or Content of the note are updated.</p>

Field	Details
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable Sort</p> <p>Description Date the note was last viewed. This field is available in API version 35.0 and later.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the owner of the note.</p>
TextPreview	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A preview of the note's content. This field is available in API version 35.0 and later.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Namefield, Sort, Update</p> <p>Description Title of the note.</p>

Usage

- The ContentNote object represents notes created with the enhanced note taking tool, released in Winter '16.
- Use ContentNote to create, query, retrieve, search, edit, and update notes.
- ContentNote is built on ContentVersion, and so it has many of the same usages.
- Not all fields can be set for notes. Only the `Content` and `Title` fields can be updated.
- The maximum file size you can upload via the SOAP API must be less than 50 MB. When a document is uploaded or downloaded via the API, it is converted to base64. This conversion increases the document size by approximately 37%. Account for the base64 conversion increase so that the file you plan to upload is less than 50 MB.
- SOQL and SOSL queries on the ContentNote return only the most recent version of the note.
- To relate a note to a record, use [ContentDocumentLink](#).

For example, the following code creates a note and escapes any special characters so they are converted to their HTML equivalents.

```
ContentNote cn = new ContentNote();
cn.Title = 'test1';
String body = 'Hello World. Before insert/update, escape special characters such as ", ', '&, and other standard escape characters.';
cn.Content = Blob.valueOf(body.escapeHTML4());
insert(cn);
```

In this example, the following code creates a note using text that is already formatted as HTML, so it does not need to be escaped.

```
ContentNote cn = new ContentNote();
cn.Title = 'test2';
String body = '<b>Hello World. Because this text is already formatted as HTML, it does not
need to be escaped.
Special characters such as &quot;, etc. must already use their HTML equivalents.</b>';
cn.Content = body;
insert(cn);
```

ContentVersion

Represents a specific version of a document in Salesforce CRM Content or Salesforce Files. This object is available in versions 17.0 and later for Salesforce CRM Content documents. This object is available in versions 20.0 and later for Salesforce Files.

The maximum number of versions that can be published in a 24-hour period is 36,000.

Supported Calls

`create(), describeLayout(), describeSObjects(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

- Customer and Partner Portal users must have the “View Content in Portal” permission to query content in libraries where they have access.
- Customer and Partner Portal users can only publish, version, or edit documents if they have a Salesforce CRM Content feature license.
- All users with a content feature license can create versions in their personal library.
- Users (including users with the “View All Data” permission) can only query files they have access to, including:
 - All Salesforce CRM Content files in libraries they’re a member of and in their personal library, regardless of library permissions (API version 17.0 and later).
 - All Salesforce Files they own, posted on their profile, posted on groups they can see, and shared directly with them (API version 21.0 and later).
- All users can update versions in their personal library.
- The owner of a version or document can update the document if they are a member of the library, regardless of library permissions.
- To update a Salesforce CRM Content document, the user must be a member of the library with one of these library privileges enabled:
 - “Add Content”
 - “Add Content On Behalf of Others”
 - “Manage Library”

- `FileType` is defined by either `ContentUrl` for links or `PathOnClient` for documents, but not both.
- In API version 34.0 and later, any file can be shared with libraries, whether the file originated in Chatter or in Salesforce CRM Content.

Fields

Field	Details
Checksum	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description MD5 checksum for the file.</p>
ContentDocumentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the document.</p>
ContentLocation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Origin of the document. Valid values are: <ul style="list-style-type: none"> • S—This is a document located within Salesforce. Label is Salesforce. • E—This is a document located outside of Salesforce. Label is External. </p>
ContentModifiedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user who modified the document.</p>
ContentModifiedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description Date the document was modified.</p>

Field	Details
	ContentModifiedDate updates when, for example, the document is renamed or a new document version is uploaded. When uploading the first version of a document, ContentModifiedDate can be set to the current time or any time in the past.
ContentSize	Type int
	Properties Filter, Group, Nillable, Sort
	Description Size of the document in bytes. Always zero for links.
ContentUrl	Type url
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description URL for links. This is only set for links. One of the fields that determines the FileType. The character limit in API versions 38.0 and later is 1,300. The character limit in API versions 32.0 and earlier was 255.
Description	Type textarea
	Properties Create, Filter, Nillable, Sort, Update
	Description Description of the content version.
Division	Type picklist
	Properties Defaulted on create, Filter, Group, Restricted picklist, Sort
	Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.
ExternalDataSourceId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
Description	
ID of the external document referenced in the <code>ExternalDataSource</code> object.	
ExternalDocumentInfo1	Type string
Properties	
Create, Filter, Nillable, Sort, Update	
Description	
Stores the URL of the file in the external content repository. The integration from the external source determines the content for this string. After the reference or copy is created, the URL of the external file is updated when you:	
<ul style="list-style-type: none"> • Republish a file reference in Lightning Experience • Open the document • Create a file reference in the Chatter REST API with <code>reuseReference</code> set to true. 	
When the file is updated, the shared link is updated to the most current version.	
ExternalDocumentInfo2	Type string
Properties	
Create, Filter, Nillable, Sort, Update	
Description	
Contains the external file ID. Salesforce determines the content for this string, which is private. The content can change without notice, depending on the external system. After the file reference is created, this field isn't updated, even if the file path changes.	
FeaturedContentBoost	Type int
Properties	
Filter, Group, Nillable, Sort	
Description	
Read only. Designates a document as featured.	
FeaturedContentDate	Type date
Properties	
Filter, Group, Nillable, Sort	
Description	
Date the document was featured.	
FileExtension	Type string

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description File extension of the document. This field is available in API version 31.0 and later.</p>
FileType	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Type of content determined by <code>ContentUrl</code> for links or <code>PathOnClient</code> for documents.</p>
FirstPublishLocationId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the location where the version was first published. If the version is first published into a user's personal library or My Files, the field will contain the ID of the user who owns the personal library or My Files. If the first version is published into a public library, the field will contain the ID of that library. This field is only set the first time a version is published via the API. It isn't set when a version is published in Salesforce CRM Content or uploaded in Salesforce Files or. After the version is published, it is a read-only field. If you don't set a <code>FirstPublishLocationId</code> via the API, this field defaults to the user's personal library, unless <code>Origin</code> is set to <code>H</code>, then this field defaults to My Files.</p>
IsAssetEnabled	<p>Type boolean</p> <p>Properties Create, Group, Defaulted on create</p> <p>Description Can be specified on insert of ContentVersion to automatically convert a ContentDocument file into a ContentAsset. This field can be SOQL queried, but it can't be edited. This field is available in API version 38.0 and later.</p>
IsEncrypted	 Note: This information is about Shield Platform Encryption and not Classic Encryption. <p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether files are encrypted using Shield Platform Encryption (<code>true</code>) or not (<code>false</code>). This field is available in API version 34.0 and later.</p>
IsLatest	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the latest version of the document (<code>true</code>) or not (<code>false</code>).</p>
IsMajorVersion	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> if the document is a major version; <code>false</code> if the document is a minor version. Major versions can't be replaced.</p>
Language	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Restricted picklist, Update</p> <p>Description The language for this document. This field defaults to the user's language unless the organization is multi-language enabled. Specifies the language of the labels returned. The value must be a valid user locale (language and country), such as <code>de_DE</code> or <code>en_GB</code>. For more information on locales, see the Language field on the CategoryNodeLocalization object.</p>
NegativeRatingCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read only. The number of times different users have given the document a thumbs down. Rating counts for the latest version are not version-specific. If Version 1 receives 10 thumbs-down votes, and Version 2 receives 2 thumbs-down votes, the <code>NegativeRatingCount</code> on Version 2 is 12. However, rating counts are not retroactive for prior versions. The <code>NegativeRatingCount</code> on Version 1 is 10.</p>

Field	Details
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the community that this file originated from. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. You can only add a <code>NetworkId</code> when creating a file. You can't change or add a <code>NetworkId</code> for an existing file.</p>
Origin	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The source of the content version. Valid values are:</p> <ul style="list-style-type: none"> • C—This is a Content document from the user's personal library. Label is Content. The <code>FirstPublishLocationId</code> must be the user's ID. If <code>FirstPublishLocationId</code> is left blank, it defaults to the user's ID. • H—This is a Salesforce file from the user's My Files. Label is Chatter. The <code>FirstPublishLocationId</code> must be the user's ID. If <code>FirstPublishLocationId</code> is left blank, it defaults to the user's ID. Origin can only be set to H if Chatter is enabled for your organization. <p>This field defaults to C. Label is Content Origin.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of this document.</p>
PathOnClient	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description The complete path of the document. One of the fields that determines the <code>FileType</code>.</p> <p> Note: You must specify a complete path including the path extension in order for the document to be visible in the Preview tab.</p>

Field	Details
PositiveRatingCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read only. The number of times different users have given the document a thumbs up. Rating counts for the latest version are not version-specific. If Version 1 receives 10 thumbs-up votes, and Version 2 receives 2 thumbs-up votes, the PositiveRatingCount on Version 2 is 12. However, rating counts are not retroactive for prior versions. The PositiveRatingCount on Version 1 is 10.</p>
PublishStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Indicates if and how the document is published. Valid values are:</p> <ul style="list-style-type: none"> • P—The document is published to a public library and is visible to other users. Label is Public. • R—The document is published to a personal library and is not visible to other users. Label is Personal Library. • U—The document is not published because publishing was interrupted. Label is Upload Interrupted.
RatingCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read only. Total number of positive and negative ratings.</p>
ReasonForChange	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The reason why the document was changed. This field can only be set when inserting a new version (revising) a document.</p>
RecordTypeID	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type of the version. Custom fields are restricted in <code>RecordTypeId</code>. When an administrator creates a custom field via the API it must be added to at least one page layout:</p> <ul style="list-style-type: none"> • If the custom field is added to the page layout associated with the General record type, the <code>RecordTypeId</code> that corresponds to that record type does not have to be set on the version record. • If the custom field is added to the page layout associated with a custom record type, the <code>RecordTypeId</code> that corresponds to that record type must be set on the version record.
SharingOption	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Controls whether or not sharing is frozen for a file. Only administrators and file owners with Collaborator access to the file can modify this field. Default is <code>Allowed</code>, which means that new shares are allowed. When set to <code>Restricted</code>, new shares are prevented without affecting existing shares. This field is available in API versions 35.0 and later.</p>
TagCsv	<p>Type textarea</p> <p>Properties Create, Nillable, Sort, Update</p> <p>Description Text used to apply tags to a content version via the API.</p>
TextPreview	<p>Type string</p> <p>Properties Nillable, Filter, Group, Sort</p> <p>Description A preview of a document. Available in API version 35.0 and later.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
Description	The title of a document.
VersionData	Type base64
Properties	Create, Nillable, Update
Description	Encoded file data. This field can't be set for links. The maximum file size you can upload via the SOAP API must be less than 50 MB. When a document is uploaded or downloaded via the API, it is converted to base64 and stored in <code>VersionData</code> . This conversion increases the document size by approximately 37%. You must account for the base64 conversion increase so that the file you plan to upload is less than 50 MB.
VersionNumber	Type string
Properties	Filter, Group, Nillable, Sort
Description	The version number. The number increments with each version of the document, for example, 1, 2, 3.

Usage

- Use this object to create, query, retrieve, search, edit, and update a specific version of a Salesforce CRM Content document or Salesforce file. Use the `ContentDocument` object to retrieve, query, update, and delete the latest version of a document, but not a content pack, in a library or a Salesforce file .
- Use this object to create, query, retrieve, search, edit, and update a specific version of a Salesforce file. Use the `ContentDocument` object to retrieve, query, update, and delete the latest version of a Salesforce file .
- To query a file that is shared only with a record, you must specify the content ID of the file.
- Not all fields can be set for Salesforce Files.
- You can only update a version if it is the latest version and if it is published.
- You can't archive versions.
- Using API version 32.0 and later, you can update record types on versions.
- You can't delete a version via the API.
- The maximum file size you can upload via the SOAP API must be less than 50 MB. When a document is uploaded or downloaded via the API, it is converted to base64 and stored in `VersionData`. This conversion increases the document size by approximately 37%. You must account for the base64 conversion increase so that the file you plan to upload is less than 50 MB.
- To download a document via the API, you must export the `VersionData` of the document. This does not increase the download count.

- When you upload a document from your local drive using the Data Loader, you must specify the actual path in both `VersionData` and `PathOnClient`. `VersionData` identifies the location and extracts the format and `PathOnClient` identifies the type of document being uploaded.
- SOQL queries on the ContentVersion object return all versions of the document. SOSL searches on the ContentVersion object return only the most recent version of the document.
- If you query versions in the API, versions with a `PublishStatus` of `Upload Interrupted` are not returned.
- Documents published into a personal library assume the default record type that is set for the user profile of the person publishing the document (General, if no default is set for the user profile).



Note: An administrator can rename the default (*Content Version Layout*) page layout.

- Contact Manager, Group, Professional, Enterprise, Unlimited, and Performance Edition customers can publish a maximum of 36,000 new versions per 24-hour period. Developer Edition and trial users can publish a maximum of 2,500 new versions per 24-hour period.
- Custom validation rules can prevent an update of documents published into a personal library via the API.

Applying Tags to ContentVersion Records

Tags can be applied to ContentVersion records using either Enterprise or Partner API.

To apply tags to a ContentVersion record, set a value in the `TagCsv` field. For example, setting this field to `one, two, three` creates and associates three tags to that version.

- The maximum length of the `TagCsv` field is 2,000 characters.
- The maximum length of an individual tag is 100 characters.
- When tags are applied to a version, the content is indexed automatically and the tags are searchable.
- You can't apply tags to a `TagCsv` that is published into a personal library.
- You can't apply tags using the ContentDocument object.
- You can't change or delete tag names. You can remove tags from a document, but that doesn't delete the tag.
- Tags are case insensitive. You can't have two tags with the same name even if they use different uppercase and lowercase letters. The case of the original tag is always used.

To delete tags from a ContentVersion record, perform a standard API update, and remove any values from the `TagCsv` field that you want to delete. For example, if the original `TagCsv` is `one, two, three`, perform an API update specifying `one, three` in the `TagCsv` field to delete `two`. To delete all tags from a ContentVersion you perform a standard API update by setting the field to `null`.

If you create a ContentVersion record and want to revise it via the API, you insert another ContentVersion record but associate it to the same ContentDocument record as the original. This has an impact on tagging:

- If you insert the revision and do not set any value in the `TagCsv` field, any tags applied to the previous version are automatically applied to the new version.
- If you insert the revision and specify a new `TagCsv` field, no tags transfer over and the tags you specify are applied instead.

When you perform a SOQL query for a ContentVersion record and select the `TagCsv` field, all the tags associated with that record are returned. The tags in the string are always ordered alphabetically even if they were inserted in a different order. You can't use the `TagCsv` field as part of a filter in a SOQL query. You can't query all tags in your organization.

Library tagging rules:

- API tagging respects the tagging restrictions that exist on any library that the document is published into. For example, if the library is in restricted tagging mode and only allows tags `one, three`, you can't save a version with a `TagCsv` of `one, two, three`.

- If the library is in guided tagging mode, you can apply tags to the ContentVersion. You can't query the value of guided tags on a library, but you can query the tagging model of a library.

SEE ALSO:

[ContentDocument](#)
[ContentVersionHistory](#)

ContentVersionHistory

Represents the history of a specific version of a document. This object is available in version 17.0 and later.

Supported Calls

`describeObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

- Customer and Partner Portal users must have the "View Content in Portal" permission to query content in libraries where they have access.
- A user can query all versions of a document from their personal library and any version that is part of or shared with a library where they are a member, regardless of library permissions.

Fields

Field	Details
ContentVersionId	Type reference Properties Filter, Group, Sort Description ID of the version.
Division	Type picklist Properties Defaulted on create, Filter, Group, Restricted picklist, Sort Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed. Possible values include:</p> <ul style="list-style-type: none"> • contentVersionCreated—A new version is created. • contentVersionUpdated—The title, description, or any custom field on the version is changed. • contentVersionDownloaded—A version is downloaded. • contentVersionViewed—The version details are viewed. • contentVersionRated—The version is rated. • contentVersionCommented—The version receives a comment. • contentVersionDataReplaced—The new version replaces the previous version, which can happen only when the new version is uploaded immediately after the previous version.
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this read-only object to query the history of a document version.

SEE ALSO:

[ContentVersion](#)

ContentWorkspace

Represents a public library in Salesforce CRM Content. This object is available in versions 17.0 and later.

 **Note:** This object does not apply to personal libraries.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

- Customer and Partner Portal users can only edit the library document object if they have a Salesforce CRM Content feature license.
- Customer and Partner Portal users can query this object if they have the “View Content in Portal” permission. A user can query all public libraries where they are members, regardless of library permissions.

Fields

Field	Details
DefaultRecordTypeId	Type reference Properties Filter, Nullable Description ID of the default content type for the library. Content types are the containers for custom fields in Salesforce CRM Content.
Description	Type textarea Properties Filter, Nullable Description Text description of the content library.
IsRestrictContentTypes	Type boolean Properties Defaulted on create, Filter Description Read only. Indicates whether content types have been restricted (<code>true</code>) or not (<code>false</code>).

Field	Details
IsRestrictLinkedContentTypes	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Read only. Indicates whether linked content types have been restricted (<code>true</code>) or not (<code>false</code>).</p>
Name	<p>Type string</p> <p>Properties Filter, idLookup</p> <p>Description Name of the library.</p>
TagModel	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Nullable, Restricted picklist</p> <p>Description The type of tagging assigned to a library. Valid values are:</p> <ul style="list-style-type: none"> • U — Unrestricted. No restrictions on tagging. Users can enter any tag when publishing or editing content. • G — Guided. Users can enter any tag when publishing or editing content, but they are also offered a list of suggested tags. • R — Restricted. Users must choose from a list of suggested tabs.

Usage

Use this object to query libraries to find out where documents can be published.

If the content type is not specified when publishing a new version into a library, it will be determined by the `DefaultRecordTypeID` of the primary library.

You can't create, update, or delete a library via the API.

SEE ALSO:

[ContentWorkspaceDoc](#)

ContentWorkspaceDoc

Represents a link between a document and a public library in Salesforce CRM Content. This object is available in versions 17.0 and later.

 **Note:** This object does not apply to documents and versions in a personal library.

Supported Calls

`create(), delete(), describeObjects() query(), retrieve(), update(), upsert()`

Special Access Rules

- Customer and Partner Portal users must have the “View Content in Portal” permission in order to query and obtain content in libraries where they have access.
- Customer and Partner Portal users can only edit documents if they have a Salesforce CRM Content feature license.
- To create a ContentWorkspaceDoc, you must be a member of the library with one of these library privileges enabled:
 - “Add Content”
 - “Add Content On Behalf of Others”
 - “Manage Library”
- To query all library documents in a library, a user must be a member of that library, regardless of library permissions.

Fields

Field	Details
ContentDocumentId	Type reference Properties Create, Filter Description Read only. ID of the library document.
ContentWorkspaceId	Type reference Properties Create, Filter Description Read only. ID of the library.
IsOwner	Type boolean Properties Defaulted on create, Filter Description Read only. Indicates whether the library owns the document and determines permissions for that document (<code>true</code>) or not (<code>false</code>). Documents can belong to

Field	Details
	more than one library, but only one library owns the document and determines its permissions.

Usage

- Use this object to link a document to one or more libraries.
- To share a document with additional libraries, create additional ContentWorkspaceDoc records which join the document to the additional libraries.
- Inserting a ContentWorkspaceDoc triggers the publish process for public libraries.
- A document can be published into many public libraries, but it will always be owned by one library which controls the security of the document.
- A document can only be published into the document owner's personal library. You can't publish into another user's personal library. Personal libraries are not visible via the API.
- To publish a document into a personal library, you must specify your user ID as the first publish location ID. If you leave the first publish location ID blank, it defaults to the current user's ID.
- A document can be published from a personal library into a public library, but once it has been published into the public library, it can't be published into the personal library again.
- You can't publish a document from a personal library into a public library that has restricted content types.
- You can't update or delete a library document via the API.

SEE ALSO:

[ContentWorkspace](#)

Contract

Represents a contract (a business agreement) associated with an Account.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	Description
	Required. ID of the Account associated with this contract.
ActivatedById	Type reference
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	ID of the User who activated this contract.
ActivatedDate	Type dateTime
	Properties
	Filter, Nillable, Sort, Update
	Description
	Date and time when this contract was activated.
BillingAddress	Type address
	Properties
	Filter, Nillable
	Description
	The compound form of the billing address. Read-only. See Address Compound Fields for details on compound address fields.
BillingCity	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Details for the billing address. Maximum size is 40 characters.
BillingCountry	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Details for the billing address of this account. Maximum size is 80 characters.
BillingCountryCode	Type picklist

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the contract's billing address.</p>
BillingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>
BillingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
BillingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address of this account. Maximum size is 20 characters.</p>
BillingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Details for the billing address. Maximum size is 80 characters.</p>
BillingStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the contract's billing address.</p>

Field	Details
BillingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street address for the billing address.</p>
CompanySignedDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date on which the contract was signed by your organization.</p>
CompanySignedId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the User who signed the contract.</p>
ContractNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description Number of the contract.</p>
ContractTerm	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Number of months that the contract is valid.</p>
CustomerSignedDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date on which the customer signed the contract.</p>

Field	Details
CustomerSignedId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the Contact who signed this contract.</p>
CustomerSignedTitle	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Title of the customer who signed the contract.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the contract.</p>
EndDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort,</p> <p>Description Read-only. Calculated end date of the contract. This value is calculated by adding the ContractTerm to the StartDate.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LastActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description
	Value is one of the following, whichever is the most recent:
	<ul style="list-style-type: none"> • Due date of the most recent event logged against the record. • Due date of the most recently closed task associated with the record.
LastApprovedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	Last date the contract was approved.
LastReferencedDate	Type
	date
	Properties
	Filter, Nillable, Sort, Update
	Description
	The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type
	date
	Properties
	Filter, Nillable, Sort, Update
	Description
	The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
OwnerExpirationNotice	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	Number of days ahead of the contract end date (15, 30, 45, 60, 90, and 120). Used to notify the owner in advance that the contract is ending.
OwnerId	Type
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	ID of the user who owns the contract.

Field	Details
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the pricebook, if any, associated with this contract.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
ShippingAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the shipping address. Read-only. See Address Compound Fields for details on compound address fields.</p>
ShippingCity	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Details of the shipping address. City maximum size is 40 characters.</p>
ShippingCountry	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Details of the shipping address. Country maximum size is 80 characters.</p>
ShippingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	Description
	The ISO country code for the contract's shipping address.
ShippingLatitude	Type
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address.
	Acceptable values are numbers between -90 and 90 with up to 15 decimal places.
ShippingLongitude	Type
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address.
	Acceptable values are numbers between -180 and 180 with up to 15 decimal places.
ShippingPostalCode	Type
	string
	Properties
	Create, Filter, Nillable, Update
	Description
	Details of the shipping address. Postal code maximum size is 20 characters.
ShippingState	Type
	string
	Properties
	Create, Filter, Nillable, Update
	Description
	Details of the shipping address. State maximum size is 80 characters.
ShippingStateCode	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The ISO state code for the contract's shipping address.
ShippingStreet	Type
	string

Field	Details
	<p>Properties Create, Filter, Nillable, Update</p> <p>Description The street address of the shipping address. Maximum of 255 characters.</p>
SpecialTerms	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Special terms that apply to the contract.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Start date for this contract. Label is Contract Start Date.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The picklist of values that indicate order status. Each value is within one of two status categories defined in <code>StatusCode</code>. For example, the status picklist may contain: Ready to Ship, Shipped, Received as values within the Activated <code>StatusCode</code>.</p>
StatusCode	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The status category for the contract. A contract can be Draft, InApproval, or Activated. Label is Status Category.</p>

Usage

The Contract object represents a business agreement.

The `Status` field specifies the current state of a contract. Status strings (defined in the `ContractStatus` object) represent its current state (Draft, InApproval, or Activated).

Client applications must initially create a Contract in a non-Activated state. Client applications can subsequently activate a Contract by updating it and setting the value in its `Status` field to Activated; however, the `Status` field is the only field you can update when activating the Contract.

Once a Contract has been activated, your client application can't change its status; however, prior to activation, your client application can change the status value from Draft to InApproval via the API. Also, your client application can delete contracts whose status is Draft or InApproval but not when a contract status is Activated.

Client applications can use the API to create, update, delete, and query any Attachment associated with a contract.

SEE ALSO:

[ContractContactRole](#)

[ContractHistory](#)

[ContractStatus](#)

ContractContactRole

Represents the role that a given Contact plays on a Contract.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the Contact associated with this Contract.</p>
ContractId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Contract.</p>
IsDeleted	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Specifies whether this Contact plays the primary role on this Contract (<code>true</code>) or not (<code>false</code>). Note that each contract has only one primary contact role. Default is <code>false</code>. Labels is Primary.</p>
Role	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Group, Sort, Update</p> <p>Description Name of the role played by the Contact on this Contract, such as Decision Maker, Approver, Buyer, and so on. Must be unique—there can't be multiple records in which the <code>ContractId</code>, <code>ContactId</code>, and <code>Role</code> values are identical. Different contacts can play the same role on the same contract. A contact can play different roles on the same contract.</p>

Usage

Use the `ContractContactRole` object to define the role that a given Contact plays on a given Contract within the context of a specific Opportunity.

SEE ALSO:

[ContractStatus](#)

ContractFeed

Represents a single feed item in the feed on the contract record detail page. This object is available in API version 18.0 and later.

A contract feed shows recent changes to a contract record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to contracts in Salesforce.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Contract object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of ContractFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>

Field	Details
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>

Field	Details
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field.</p> <p>When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>.</p> <p>Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>

Field	Details
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the contract record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>

Field	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.

Field	Details
	<ul style="list-style-type: none"> • <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed. • <code>LinkPost</code>—a post with an attached URL. • <code>PollPost</code>—a poll posted on a feed. • <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile. • <code>QuestionPost</code>—generated when a user posts a question. • <code>ReplyPost</code>—generated when Chatter Answers posts a reply. • <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Visibility

Type
picklist

Properties

Filter, Group, Nillable, Restricted picklist, Sort

Description

Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.

`Visibility` can have the following values:

- `AllUsers`—The feed item is available to all users who have permission to see the feed item.
- `InternalUsers`—The feed item is available to internal users only.

Note the following exceptions for `Visibility`:

- For record posts, `Visibility` is set to `InternalUsers` for all internal users by default.
- External users can set `Visibility` only to `AllUsers`.

Field	Details
	<ul style="list-style-type: none"> On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a contract record.

SEE ALSO:

[Contract](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

ContractHistory

Represents the history of changes to the values in the fields of a contract.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
ContractId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Contract. Label is Contract ID.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to a contract.

This object respects field level security on the parent object.

SEE ALSO:

[ContractStatus](#)

ContractLineItem

Represents a Product2 in a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
AssetId	<p>Type reference</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Required. ID of the Asset associated with the contract line item. Must be a valid asset ID.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description Description of the contract line item.</p>
Discount	<p>Type percent</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The discount for the product as a percentage. When updating, if you specify <code>Discount</code> without specifying <code>TotalPrice</code>, the <code>TotalPrice</code> will be adjusted to accommodate the new <code>Discount</code> value, and the <code>UnitPrice</code> will be held constant. If you specify both <code>Discount</code> and <code>Quantity</code>, you must also specify either <code>TotalPrice</code> or <code>UnitPrice</code> so the system can determine which one to automatically adjust.</p>
EndDate	<p>Type date</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The last day the contract line item is in effect.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nullable, Sort, Update</p>

Field	Details
LastViewedDate	<p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LineItemNumber	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p>
ListPrice	<p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
ParentContractLineItemId	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Update</p>
PricebookEntryId	<p>Description Automatically-generated number that identifies the contract line item.</p>
PricebookEntryId	<p>Type currency</p> <p>Properties Filter, Nillable</p>
ParentContractLineItemId	<p>Description Corresponds to the <code>UnitPrice</code> on the PricebookEntry that is associated with this line item, which can be in the standard pricebook or a custom pricebook. A client application can use this information to show whether the unit price (or sales price) of the line item differs from the pricebook entry list price.</p>
PricebookEntryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
PricebookEntryId	<p>Description The line item's parent line item, if it has one.</p>
PricebookEntryId	<p>Type reference</p> <p>Properties Create, Filter</p>
PricebookEntryId	<p>Description Required. ID of the associated PricebookEntry. Only exists if Product2 is enabled.</p>

Field	Details
Quantity	<p>Type double</p> <p>Properties Create, Filter, Update</p> <p>Description Number of units of the contract line item (product) included in the associated service contract.</p>
RootContractLineItemId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The top-level line item in a contract line item hierarchy. Depending on where a line item lies in the hierarchy, its root could be the same as its parent.</p>
ServiceContractId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the ServiceContract associated with the contract line item. Must be a valid asset ID.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The first day the contract line item is in effect.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Nillable</p> <p>Description Status of the contract line item.</p>
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable</p>

Field	Details
	<p>Description Contract line item's sales price multiplied by the <code>Quantity</code>.</p>
TotalPrice	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description This field is available only for backward compatibility. It represents the total price of the ContractLineItem If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>UnitPrice</code> is required. This field is nillable, but you can't set both <code>TotalPrice</code> and <code>UnitPrice</code> to null in the same update request. To insert the <code>TotalPrice</code> for a contract line item via the API (given only a unit price and the quantity), calculate this field as the unit price multiplied by the quantity.</p>
UnitPrice	<p>Type currency</p> <p>Properties Create, Filter, Update</p> <p>Description The unit price for the contract line item. In the user interface, this field's value is calculated by dividing the total price of the contract line item by the quantity listed for that line item. Label is Sales Price. This field or <code>TotalPrice</code> is required. You can't specify both. If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>TotalPrice</code> is required.</p>

Usage

Use this object to query and manage contract line items.

SEE ALSO:

[ContractLineItemHistory](#)

ContractLineItemHistory

Represents the history of changes to the values in the fields on a ContractLineItem (items in a customer support agreement). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
ContractLineItemId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. ID of the ContractLineItem.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field	Details
	<p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to a contract line item.

This object respects field level security on the parent object.

SEE ALSO:

[ContractLineItem](#)

ContractStatus

Represents the status of a Contract, such as Draft, InApproval, Activated, Terminated, or Expired.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
IsDefault	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the default contract status value (<code>true</code>) or not (<code>false</code>) in the picklist.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this contract status value. This display value is the internal label that does not get translated.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the contract status picklist. These numbers are not guaranteed to be sequential, as some previous contract status values might have been deleted.</p>
StatusCode	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Code indicating the status of a contract. One of the following values:</p> <ul style="list-style-type: none"> • Draft • InApproval • Activated <p>Two other values (<code>Terminated</code> and <code>Expired</code>) are defined but are not available for use via the API.</p>

Usage

This object represents a value in the contract status picklist. The contract status picklist provides additional information about the status of a Contract, such as its current state (`Draft`, `InApproval`, or `Activated`). You can query these records to retrieve the set of values in the contract status picklist, and then use that information while processing Contract objects to determine more information

about a given contract. For example, the application could test whether a given contract is activated based on its `Status` value and the value of the `StatusCode` property in the associated `ContractStatus` object.

SEE ALSO:

[ContractContactRole](#)

ContractTag

Associates a word or short phrase with a Contract.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Restricted picklist</p>
	<p>Description Defines the visibility of a tag.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching <code>OwnerId</code>.

Usage

ContractTag stores the relationship between its parent TagDefinition and the Contract being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

CronJobDetail

Contains details about the associated scheduled job, such as the job's name and type. This object is available in API version 29.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
JobType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of the associated scheduled job. The following are the available job types. Each job type label is listed with its value in parenthesis. Use the job type value when querying for a specific job type.</p> <ul style="list-style-type: none"> • Data Export (0) • Dashboard Refresh (3) • Reporting Snapshot (4)

Field	Details
	<ul style="list-style-type: none"> Scheduled Apex (7) Report Run (8) Batch Job (9)
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort,</p> <p>Description The name of the associated scheduled job.</p>

Usage

Use this object to query additional information about a scheduled job, such as the job's name and type.

CronTrigger

Contains schedule information for a scheduled job. CronTrigger is similar to a cron job on UNIX systems. This object is available in API version 17.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
CronExpression	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The cron expression used to initiate the schedule.</p> <p>Syntax:</p> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> <code>Seconds Minutes Hours Day_of_month Month Day_of_week Optional_year</code> </div> <p>See schedule(jobName, cronExpression, schedulableClass) in the <i>Apex Developer Guide</i>.</p>

Field	Details
CronJobDetailId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the CronJobDetail record containing more details about this scheduled job.</p>
EndTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time when the job either finished or will finish.</p>
NextFireTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The next date and time the job is scheduled to run. null if the job is not scheduled to run again.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Owner of the job.</p>
PreviousFireTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The most recent date and time the job ran. null if the job has not run before current local time.</p>
StartTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description The date and time when the most recent iteration of the scheduled job started.</p>
State	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The current state of the job. The job state is managed by the system. Possible values are:</p> <ul style="list-style-type: none"> • WAITING—The job is waiting for execution. • ACQUIRED—The job has been picked up by the system and is about to execute. • EXECUTING—The job is executing. • COMPLETE—The trigger has fired and is not scheduled to fire again. • ERROR—The trigger definition has an error. • DELETED—The job has been deleted. • PAUSED—A job can have this state during patch and major releases. After the release has finished, the job state is automatically set to WAITING or another state. • BLOCKED—Execution of a second instance of the job is attempted while one instance is running. This state lasts until the first job instance is completed. • PAUSED_BLOCKED—A job has this state due to a release occurring. When the release has finished and no other instance of the job is running, the job's status is set to another state.
TimesTriggered	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of times this job has been triggered.</p>
TimeZoneSidKey	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Returns the timezone ID. For example, America/Los_Angeles.</p>

Usage

Use this object to query scheduled jobs in your organization.

CurrencyType

Represents the currencies used by an organization for which the multicurrency feature is enabled.

Supported Calls

`create()`, `describeObjects()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

- This object is not available in single-currency organizations.
- You need the “Customize Application” permission to edit this object.
- Your client application can't delete this object.
- Customer Portal users can't access this object.

Fields

Field	Details
ConversionRate	Type double Properties Filter Description Required. Conversion rate of this currency type against the corporate currency.
DecimalPlaces	Type int Properties Filter Description Required. For this currency, specifies the number of digits to the right of the decimal point, such as zero (0) for JPY or 2 for USD.
IsActive	Type boolean Properties Defaulted on create, Filter Description Indicates whether this currency type is active (<code>true</code>) or not (<code>false</code>). Inactive currency types do not appear in picklists in the user interface. Label is Active . This field defaults to <code>false</code> if no value is provided when updating or inserting a record.

Field	Details
IsCorporate	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether this currency type is the corporate currency (<code>true</code>) or not (<code>false</code>). Label is Corporate Currency. All other currency conversion rates are applied against this corporate currency. If a currency is already defined as the corporate currency in the user interface, it can't be unset. When a non-corporate currency is set to a corporate currency, the system reconfigures all conversion rates based on the new corporate currency.</p>
IsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description Required. ISO code of the currency. Must be one of the valid alphabetic, three-letter currency ISO codes defined by the ISO 4217 standard, such as <code>USD</code>, <code>GBP</code>, or <code>JPY</code>. Must be unique within your organization. Label is Currency ISO Code.</p>

Usage

This object is for multicurrency organizations only. Use this object to define the currencies your organization uses.

When updating an existing record, make sure to provide values for all fields to avoid undesired changes to the `CurrencyType`. For example, if a value for `IsActive` is not provided, the default (`false`) is used, which could result in a currently active `CurrencyType` becoming inactive.

SEE ALSO:

[DatedConversionRate](#)

[Object Basics](#)

CustomBrand

Represents a custom branding and color scheme. This object is available in API version 28.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the parent entity that this branding applies to. The parent entity can be either a network, organization, topic, or reputation level. The branding applies to the entity that the ParentId references. For example, if the ParentId references a network ID, the branding applies to that network (community) only, and if the ParentId references an organization ID, the branding applies to the organization that it is accessed through, and so on. Label is Branded Entity ID.</p>

Usage

Use this object along with [CustomBrandAsset](#) to apply a custom branding scheme to your community. The branding scheme for the community shows in both the user interface and in Salesforce1. You must have “Create and Manage Communities” to customize community branding.

You can also use this object to apply a custom branding scheme to your organization when it is accessed through Salesforce1.

SEE ALSO:

[Network](#)

CustomBrandAsset

Represents a branding element in a custom branding scheme. For example, a color, logo image, header image, or footer text. A CustomBrandAsset can apply to a community or to an organization using Salesforce1. This object is available in API version 28.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
AssetCategory	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Values include:</p> <ul style="list-style-type: none"> • MotifZeronaryColor—The background color for the header. Label is Zeronary motif color. If this CustomBrandAsset is for a network, this is the header color for the network. If it is for an organization, this is the header color when users access Salesforce1. • MotifPrimaryColor—The color used for the active tab. Label is Primary motif color. Not used for Salesforce1 branding. • MotifSecondaryColor—The color used for the top borders of lists and tables. Label is Secondary motif color. Not used for Salesforce1 branding. • MotifTertiaryColor—The background color for section headers on edit and detail pages. Label is Tertiary motif color. Not used for Salesforce1 branding. • MotifQuaternaryColor—if this CustomBrandAsset is for a network, this is the background color for network pages. If it is for an organization, this is the background color on a splash page. Label is Quaternary motif color. • MotifZeronaryComplementColor—Font color used with zeronaryColor. Label is Zeronary motif colors complement color. • MotifPrimaryComplementColor—Font color used with primaryColor. Label is Primary motif colors complement color. Not used for Salesforce1 branding. • MotifTertiaryComplementColor—Font color used with tertiaryColor. Label is Tertiary motif colors complement color. Not used for Salesforce1 branding.

Field Name	Details
	<ul style="list-style-type: none"> • MotifQuaternaryComplementColor—Font color used with quaternaryColor. Label is Quaternary motif colors complement color. <p>Not used for Salesforce1 branding.</p> <ul style="list-style-type: none"> • PageHeader—An image that appears on the header of the community pages. Can be an .html, .gif, jpg, or .png file. Label is Page Header. <p>Not used for Salesforce1 branding.</p> <ul style="list-style-type: none"> • PageFooter—An image that appears on the footer of the community pages. Must be an .html file. Label is Page Footer. <p>Not used for Salesforce1 branding.</p> <ul style="list-style-type: none"> • LoginFooterText—The text that appears in the footer of the community login page. Label is Footer text displayed on the login page. <p>Not used for Salesforce1 branding.</p> <ul style="list-style-type: none"> • LoginLogoImageId—The logo that appears on the community login page for external users. In Salesforce1, this logo also appears on the community splash page. Label is Logo image displayed on the login page. <ul style="list-style-type: none"> • LargeLogoImageId—Only used for Salesforce1. The logo that appears on the splash page when you start Salesforce1. Label is Large logo image. <ul style="list-style-type: none"> • SmallLogoImageId—Only used for Salesforce1. The logo that appears on the publisher in Salesforce1. Label is Small logo image.
CustomBrandId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the associated CustomBrand.</p>
ForeignKeyAssetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the document used if the AssetCategory is PageHeader, PageFooter, or LoginLogoImageId.</p>
TextAsset	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description Text used if the AssetCategory is LoginFooterText.</p>

Usage

Use this object to add basic branding elements—color scheme, header or footer images, login page logo, or footer text—to the branding scheme ([CustomBrand](#)) for your Network. You must have “Create and Manage Communities” to customize community branding.

If you’re using communities in Salesforce1, the loading page shows the logo.

SEE ALSO:

[Network](#)

Custom Metadata Type__mdt

Represents a custom metadata record. This object is available in API version 34.0 and later.

The object name is a variable with the syntax *Custom Metadata Type*_mdt, where *Custom Metadata Type* is the Object Name for the custom metadata type associated with the custom metadata record. For example, *PicklistUsage__mdt* represents a custom metadata record based on the *PicklistUsage* custom metadata type.

Supported Calls

`describeSObjects()`, `describeLayout()`, `query()`, `retrieve()`

Fields

Field	Details
<i>Custom Field__c</i>	<p>Type Any Type</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A custom field on the record.</p>
DeveloperName	<p>Type string</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
isProtected	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>When a custom metadata type's records are released in a managed package, access to them is limited in specific ways.</p> <ul style="list-style-type: none"> Code that's in the same managed package as custom metadata records can read the records. Code that's in the same managed package as custom metadata types can read the records that belong to that type. Code that's in a managed package that doesn't contain either the type or the protected record can't read the protected records. Code that the subscriber creates and code that's in an unmanaged package can't read the protected records. The developer can modify protected records only with a package upgrade. The subscriber can't read or modify protected records. The developer name of a protected record can't be changed after release. <p>Records that are hidden by these access rules are also unavailable to REST, SOAP, SOQL, and Setup.</p>
Label	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The custom metadata record label. This label value is always the same as the <code>MasterLabel</code> value.</p>
Language	<p>Type string</p> <p>Properties Filter, Group, restrictedPicklist, Sort</p>

Field	Details
	<p>Description The language of the custom metadata record. This value is always the default language of the developing organization.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The master label for the custom metadata record.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p>
QualifiedApiName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A concatenation of the namespace prefix and developer name. The format is <i>NamespacePrefix__DeveloperName</i>.</p>

Custom Object__Feed

Represents a single feed item on a custom object detail page. This object is available in API version 18.0 and later.

A custom object feed shows recent changes to a custom object record for fields that are tracked in feeds, and posts and comments about the object. It is a useful way to stay up-to-date with changes made to a custom object. The object name is variable and uses a *Custom Object__Feed* syntax, where *Custom Object* is the Object Name for the custom object. For example, *Offer__Feed* represents a feed object for the custom object named Offer.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of <i>Custom Object__Feed</i>. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
Description	
	<p>When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	
Type	
base64	
Properties	
Nillable	
Description	
<p>Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>	
ContentDescription	
Type	
textarea	
Properties	
Nillable, Sort	
Description	
The description of the file specified in <code>ContentData</code> .	
ContentFileName	
Type	
string	
Properties	
Group, Nillable, Sort	
Description	
<p>This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>	
ContentSize	
Type	
int	
Properties	
Group, Nillable, Sort	
Description	
<p>Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>	
ContentType	
Type	
string	
Properties	
Group, Nillable, Sort	

Field	Details
	Description
	<p>Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	Type
	dateTime
	Properties
	Defaulted on create, Filter, Sort
	Description
	Date and time when this record was created. This field is a standard system field.
	Ordering by <code>CreatedDate DESC</code> sorts the feed by the most recent feed item.
FeedPostId	Type
	reference
	Properties
	Filter , Group , Nillable , Sort
	Description
	<p>This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only.</p>
	ID of the associated FeedPost . A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	<p>ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	Type
	boolean
	Properties
	Defaulted on create , Filter , Group , Sort
	Description
	<p>Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	Type
	boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
<code>LastModifiedDate</code>	<p>Type <code>dateTime</code></p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
<code>LikeCount</code>	<p>Type <code>int</code></p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the custom object that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>

Field	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL.

Field	Details
	<ul style="list-style-type: none">• <code>PollPost</code>—a poll posted on a feed.• <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile.• <code>QuestionPost</code>—generated when a user posts a question.• <code>ReplyPost</code>—generated when Chatter Answers posts a reply.• <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com.• <code>TextPost</code>—a direct text entry on a feed.• <code>TrackedChange</code>—a change or group of changes to a tracked field.• <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none">• <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Usage

Use this object to track changes for a custom object. A record of this object type is automatically created when a user enables feed tracking for a custom object.

CustomPermission

Represents a permission created to control access to a custom process or app, such as sending email. This object is available in API version 31.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A description of the custom permission. Limit: 255 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the custom permission in the API. This name can contain only underscores and alphanumeric characters and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. The label corresponds to Name in the user interface. Limit: 80 characters.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of the custom permission. Valid values are:</p> <ul style="list-style-type: none"> • Chinese (Simplified): zh_CN • Chinese (Traditional): zh_TW • Danish: da • Dutch: nl_NL • English: en_US • Finnish: fi • French: fr • German: de • Italian: it • Japanese: ja • Korean: ko

Field Name	Details
	<ul style="list-style-type: none"> Norwegian: no Portuguese (Brazil): pt_BR Russian: ru Spanish: es Spanish (Mexico): es_MX Swedish: sv Thai: th
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The custom permission label, which corresponds to Label in the user interface. Limit: 80 characters.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

Use the `CustomPermission` object to determine users' access to custom permissions.

For example, to query all permission sets where the Button1 permission is enabled:

```
SELECT Id, DeveloperName,
(select Id, Parent.Name, Parent.Profile.Name from SetupEntityAccessItems)
FROM CustomPermission
WHERE DeveloperName = 'Button1'
```

To query all permission sets and profiles with custom permissions:

```
SELECT Assignee.Name, PermissionSet.Id,
PermissionSet.Profile.Name,
PermissionSet.isOwnedByProfile,
PermissionSet.Label
FROM PermissionSetAssignment
WHERE PermissionSetId
IN (SELECT ParentId
    FROM SetupEntityAccess
    WHERE SetupEntityType =
'CustomPermission')
```

To query for all SetupEntityAccess rows with custom permissions:

```
SELECT Id, ParentId, Parent.Name, SetupEntityId
FROM SetupEntityAccess
WHERE SetupEntityType='CustomPermission'
AND ParentId
IN (SELECT Id
    FROM PermissionSet
    WHERE isOwnedByProfile = false)
```

SEE ALSO:

[CustomPermissionDependency](#)
[PermissionSet](#)
[Profile](#)
[SetupEntityAccess](#)

CustomPermissionDependency

Represents the dependency between two custom permissions when one custom permission requires that you enable another custom permission. This object is available in API version 32.0 and later.

Supported Calls

`describeLayout(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
CustomPermissionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the custom permission that requires the permission that's specified in RequiredCustomPermissionId.</p>
RequiredCustomPermissionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the custom permission that must be enabled when CustomPermissionId is enabled.</p>

Usage

The following Apex class contains a method that returns the IDs of all custom permissions that are required for the given custom permission ID. To use this class, save it in your organization.

```
public class CustomPermissionUtil {
    public String[] getAllRequiredCustomPermissions(String customPermId) {
        return getAllRequiredHelper(new String[]{customPermId});
    }

    private String[] getAllRequiredHelper(String[] customPerms) {
        CustomPermissionDependency[] requiredPerms = [SELECT RequiredCustomPermissionId
                                                       FROM CustomPermissionDependency
                                                       WHERE CustomPermissionId
                                                       IN :customPerms];
        String[] requiredPermsIds = new String[]{};
        for (CustomPermissionDependency cpd : requiredPerms) {
            requiredPermsIds.add(cpd.RequiredCustomPermissionId);
        }
        if (requiredPermsIds.size() > 0) {
            customPerms.addAll(getAllRequiredHelper(requiredPermsIds));
            return customPerms;
        } else {
            return customPerms;
        }
    }
}
```

For more information about using Apex classes, see the [Force.com Apex Code Developer's Guide](#).

SEE ALSO:

[CustomPermission](#)

DandBCompany

Represents a Dun & Bradstreet® company record, which is associated with an account added from Data.com. This object is available in API version 25.0 and later.

 **Warning:** You can update fields in the DandBCompany object; however, field changes may be overwritten by Data.com Clean jobs or by using the Data.com Clean button.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

Only organizations with Data.com Premium Prospector or Data.com Premium Clean can access this object.

Fields

Field Name	Details
Address	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city where a company is physically located. Maximum size is 40 characters.</p>
CompanyCurrencyIsoCode	<p>Type picklist</p>

Field Name	Details
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The code used to represent a company's local currency. This data is provided by the International Organization for Standardization (ISO) and is based on their three-letter currency codes. For example, USD is the ISO code for United States Dollar. Maximum size is 3 characters.
Country	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The country where a company is physically located. Maximum size is 40 characters.
CountryAccessCode	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The required code for international calls. Maximum size is 4 characters.
CurrencyCode	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The currency in which the company's sales volume is expressed. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 4 characters.
Description	Type
	textarea
	Properties
	Create, Nillable, Update
	Description
	A brief description of the company, which may include information about its history, its products and services, and its influence on a particular industry. Maximum size is 32000 characters.
DomesticUltimateBusinessName	Type
	string

Field Name	Details
DomesticUltimateDunsNumber	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The primary name of the Domestic Ultimate, which is the highest ranking subsidiary, specified by country, within an organization's corporate structure. Maximum size is 255 characters.
DunsNumber	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The D-U-N-S Number for the Domestic Ultimate, which is the highest ranking subsidiary, specified by country, within an organization's corporate structure. Maximum size is 9 characters.
EmployeeQuantityGrowthRate	Type string
	Properties Create, Filter, Group, Sort, Update
	Description The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun & Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used by industries and organizations around the world as a global standard for business identification and tracking. Maximum size is 9 characters.
EmployeesHere	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description The yearly growth rate of the number of employees in a company expressed as a decimal percentage. The data includes the total employee growth rate for the past two years.
EmployeesHere	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description The number of employees at a specified location, such as a branch location. Maximum size is 15 characters.

Field Name	Details
EmployeesHereReliability	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The reliability of the EmployeesHere figure. Available values include:</p> <ul style="list-style-type: none"> • 0—Actual number • 1—Low • 2—Estimated (for all records) • 3—Modeled (for non-US records)
EmployeesTotal	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The total number of employees in the company, including all subsidiary and branch locations. This data is only available on records that have a value of <i>Headquarters/Parent</i> in the LocationStatus field. Maximum size is 15 characters.</p>
EmployeesTotalReliability	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The reliability of the EmployeesTotal figure. Available values include:</p> <ul style="list-style-type: none"> • 0—Actual number • 1—Low • 2—Estimated (for all records) • 3—Modeled (for non-US records) <p>A blank value indicates this data is unavailable.</p>
FamilyMembers	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The total number of family members, worldwide, within an organization, including the Global Ultimate, its subsidiaries (if any), and its branches (if any). Maximum size is 5 characters.</p>

Field Name	Details
Fax	Type phone
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The company's facsimile number.
FifthNaics	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	An additional NAICS code used to further classify an organization by industry.
	Maximum size is 6 characters.
FifthNaicsDesc	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding NAICS code. Maximum size is 120 characters.
FifthSic	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
FifthSic8	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
FifthSic8Desc	Type string

Field Name	Details
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
FifthSicDesc	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
FipsMsaCode	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The Federal Information Processing Standards (FIPS) and the Metropolitan Statistical Area (MSA) codes identify the organization's location. The MSA codes are defined by the US Office of Management and Budget. Maximum size is 5 characters.
FipsMsaDesc	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's FIPS MSA code. Maximum size is 255 characters.
FortuneRank	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The numeric value of the company's Fortune 1000 ranking. A null or blank value means that the company isn't ranked as a Fortune 1000 company.
FourthNaics	Type
	string

Field Name	Details
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	An additional NAICS code used to further classify an organization by industry. Maximum size is 6 characters.
FourthNaicsDesc	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding NAICS code. Maximum size is 120 characters.
FourthSic	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
FourthSic8	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
FourthSic8Desc	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
FourthSicDesc	Type
	string
	Properties
	Create, Filter, Group, Nullable, Sort, Update

Field Name	Details
GeoCodeAccuracy	<p>Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>
GeoCodeAccuracy	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The level of accuracy of a location's geographical coordinates compared with its physical address. Available values include:</p> <ul style="list-style-type: none"> • A – <i>Non-US rooftop accuracy</i> • B – <i>Block level</i> • C – <i>Places the address in the correct city</i> • D – <i>Rooftop level</i> • I – <i>Street intersection</i> • M – <i>Mailing address level</i> • N – <i>Not matched</i> • P – <i>PO BOX location</i> • S – <i>Street level</i> • T – <i>Census tract level</i> • Z – <i>ZIP code level</i> • 0 (zero) – <i>Geocode could not be assigned</i>
GlobalUltimateBusinessName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The primary name of the Global Ultimate, which is the highest entity within an organization's corporate structure and may oversee branches and subsidiaries. Maximum size is 255 characters.</p>
GlobalUltimateDunsNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The D-U-N-S Number of the Global Ultimate, which is the highest entity within an organization's corporate structure and may oversee branches and subsidiaries. Maximum size is 9 characters.</p>

Field Name	Details
GlobalUltimateTotalEmployees	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The total number of employees at the Global Ultimate, which is the highest entity within an organization's corporate structure and may oversee branches and subsidiaries. Maximum size is 15 characters.</p>
ImportExportAgent	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Identifies whether a business imports goods or services, exports goods or services, and/or is an agent for goods. Available values include:</p> <ul style="list-style-type: none"> • A—Importer/exporter/agent • B—Importer/exporter • C—Importer • D—Importer/agent • E—Exporter/agent • F—Agent (keeps no inventory and does not take title goods) • G—None or data not available • H—Exporter
IncludedInSnP500	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A true or false value. If <code>true</code>, the company is listed in the S&P 500 Index. If <code>false</code>, the company isn't listed in the S&P 500 Index.</p>
Latitude	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Used with longitude to specify a precise location, which is then used to assess the Geocode Accuracy. Maximum size is 11 characters.</p>

Field Name	Details
LegalStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Identifies the legal structure of an organization.</p>
LocationStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Identifies the organizational status of a company. Available values are <i>Single location</i>, <i>Headquarters/Parent</i>, and <i>Branch</i>. Available values include:</p> <ul style="list-style-type: none"> • 0—Single location (no other entities report to the business) • 1—Headquarters/parent (branches and/or subsidiaries report to the business) • 2—Branch (secondary location to a headquarters location)
Longitude	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Used with latitude to specify a precise location, which is then used to assess the Geocode Accuracy. Maximum size is 11 characters.</p>
MailingAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the mailing address. Read-only. See Address Compound Fields for details on compound address fields.</p>
MailingCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city where a company has its mail delivered. Maximum size is 40 characters.</p>

Field Name	Details
MailingCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country where a company has its mail delivered. Maximum size is 40 characters.</p>
MailingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The postal code that a company uses on its mailing address. Maximum size is 20 characters.</p>
MailingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state where a company has its mail delivered. Maximum size is 20 characters.</p>
MailingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street address where a company has its mail delivered. Maximum size is 255 characters.</p>
MarketingPreScreen	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The probability that a company will pay with a significant delay compared to the agreed terms. The risk level is based on the standard Commercial Credit Score, and ranges from low risk to high risk. Available values include:</p> <ul style="list-style-type: none"> • L—<i>Low risk of delinquency</i> • M—<i>Moderate risk of delinquency</i> • H—<i>High risk of delinquency</i>

Field Name	Details
	 Important: Use this information for marketing pre-screening purposes only.
MarketingSegmentationCluster	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Twenty-two distinct, mutually exclusive profiles, created as a result of cluster analysis of Dun & Bradstreet data for US organizations.</p>
MinorityOwned	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates whether an organization is owned or controlled by a member of a minority group. Available values include:</p> <ul style="list-style-type: none"> • Y—Minority owned • N—Not minority owned
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The primary or registered name of a company. Maximum size is 255 characters.</p>
NationalId	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The identification number used in some countries for business registration and tax collection. Maximum size is 255 characters.</p>
NationalIdType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description A code value that identifies the type of national identification number used. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 5 characters.</p>
OutOfBusiness	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates whether the company at the specified address has discontinued operations. Available values include:</p> <ul style="list-style-type: none"> • 0—Unknown or not applicable • 1—Owns • 2—Rents
OwnOrRent	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates whether a company owns or rents the building it occupies. Available values include:</p> <ul style="list-style-type: none"> • Y—Out of business • N—Not out of business
ParentOrHqBusinessName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The primary name of the parent or headquarters company. Maximum size is 255 characters.</p>
ParentOrHqDunsNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The D-U-N-S Number for the parent or headquarters. Maximum size is 9 characters.</p>

Field Name	Details
Phone	Type phone
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description A company's primary telephone number.
PostalCode	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The postal code that corresponds to a company's physical location. Maximum size is 20 characters.
PremisesMeasure	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description A numeric value for the measurement of the premises.
PremisesMeasureReliability	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description A descriptive accuracy of the measurement such as actual, estimated, or modeled.
PremisesMeasureUnit	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description A descriptive measurement unit such as acres, square meters, or square feet.
PrimaryNaics	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description
	The six-digit North American Industry Classification System (NAICS) code is the standard used by business and government to classify business establishments according to their economic activity for the purpose of collecting, analyzing, and publishing statistical data related to the US business economy. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 6 characters.
PrimaryNaicsDesc	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on its NAICS code. Maximum size is 120 characters.
PrimarySic	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The four-digit Standard Industrial Classification (SIC) code is used to categorize business establishments by industry. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 4 characters.
PrimarySic8	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The eight-digit Standard Industrial Classification (SIC) code is used to categorize business establishments by industry. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 8 characters.
PrimarySic8Desc	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description
	<p>A brief description of an organization's line of business, based on its SIC code. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 80 characters.</p>
PrimarySicDesc	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>A brief description of an organization's line of business, based on its SIC code. Maximum size is 80 characters.</p>
PriorYearEmployees	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The total number of employees for the prior year.
PriorYearRevenue	Type
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The annual revenue for the prior year.
PublicIndicator	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	Indicates whether ownership of the company is public or private. Available values include:
	<ul style="list-style-type: none"> • Y—Public • N—Private
SalesTurnoverGrowthRate	Type
	double
	Properties
	Create, Filter, Nillable, Sort, Update

Field Name	Details
	<p>Description The increase in annual revenue from the previous value for an equivalent period expressed as a decimal percentage.</p>
SalesVolume	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The total annual sales revenue in the headquarters' local currency. Dun & Bradstreet tracks revenue data for publicly traded companies, Global Ultimates, Domestic Ultimates, and some headquarters.</p>
SalesVolumeReliability	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The reliability of the SalesVolume figure. Available values include:</p> <ul style="list-style-type: none"> • 0—Actual number • 1—Low • 2—Estimated (for all records) • 3—Modeled (for non-US records)
SecondNaics	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An additional NAICS code used to further classify an organization by industry. Maximum size is 6 characters.</p>
SecondNaicsDesc	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A brief description of an organization's line of business, based on the corresponding NAICS code. Maximum size is 120 characters.</p>
SecondSic	<p>Type string</p>

Field Name	Details
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
SecondSic8	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
SecondSic8Desc	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
SecondSicDesc	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
SixthNaics	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description An additional NAICS code used to further classify an organization by industry. Maximum size is 6 characters.
SixthNaicsDesc	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update

Field Name	Details
	Description
	A brief description of an organization's line of business, based on the corresponding NAICS code. Maximum size is 120 characters.
SixthSic	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
SixthSic8	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
SixthSic8Desc	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
SixthSicDesc	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
SmallBusiness	Type picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field Name	Details
	<p>Description</p> <p>Indicates whether the company is designated a small business as defined by the Small Business Administration of the US government. Available values include:</p> <ul style="list-style-type: none"> • Y—Small business site • N—Not small business site
State	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	The state where a company is physically located. Maximum size is 20 characters.
StockExchange	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	The corresponding exchange for a company's stock symbol. For example: NASDAQ or NYSE. Maximum size is 16 characters.
StockSymbol	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	The abbreviation used to identify publicly traded shares of a particular stock. Maximum size is 6 characters.
Street	<p>Type</p> <p>textarea</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	The street address where a company is physically located. Maximum size is 255 characters.
Subsidiary	<p>Type</p> <p>picklist</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field Name	Details
	<p>Description</p> <p>Indicates whether a company is more than 50 percent owned by another organization. Available values include:</p> <ul style="list-style-type: none"> • 0—Not subsidiary of another organization • 3—Subsidiary of another organization
ThirdNaics	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	An additional NAICS code used to further classify an organization by industry. Maximum size is 6 characters.
ThirdNaicsDesc	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	A brief description of an organization's line of business, based on the corresponding NAICS code. Maximum size is 120 characters.
ThirdSic	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
ThirdSic8	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update
	<p>Description</p>
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
ThirdSic8Desc	<p>Type</p> <p>string</p>
	<p>Properties</p>
	Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
ThirdSicDesc	<p>Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>
TradeStyle1	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>
TradeStyle2	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A name, different from its legal name, that an organization may use for conducting business. Similar to "Doing business as" or "DBA". Maximum size is 255 characters.</p>
TradeStyle3	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An additional tradestyle used by the organization. Maximum size is 255 characters.</p>
TradeStyle4	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An additional tradestyle used by the organization. Maximum size is 255 characters.</p>

Field Name	Details
TradeStyle5	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An additional tradestyle used by the organization. Maximum size is 255 characters.</p>
URL	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An organization's primary website address. Maximum size is 104 characters.</p>
UsTaxId	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The identification number for the company used by the Internal Revenue Service (IRS) in the administration of tax laws. Also referred to as Federal Taxpayer Identification Number. Maximum size is 9 characters.</p>
WomenOwned	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates whether a company is more than 50 percent owned or controlled by a woman. Available values include:</p> <ul style="list-style-type: none"> • Y—Owned by a woman • N—Not owned by a woman, or unknown
YearStarted	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The year the company was established or the year when current ownership or management assumed control of the company. Maximum size is 4 characters.</p>

Usage

Use this object to manage D&B Company records in your organization.

Dashboard

Represents a dashboard, which shows data from custom reports as visual components. Access is read-only. This object is available in API version 20.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `search()`

Fields

Field	Details
BackgroundDirection	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Returns the direction of the background fade. Available values are:</p> <ul style="list-style-type: none">• Top to Bottom• Left to Right• Diagonal (default value) <p>Label is Background Fade Direction.</p>
BackgroundEnd	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the ending fade color in hexadecimal. Label is Ending Color.</p>
BackgroundStart	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the starting fade color in hexadecimal. Label is Starting Color.</p>

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Returns the description of the dashboard. Limit: 255 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is <code>Dashboard Unique Name</code>.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
FolderId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. Returns the ID of the Folder that contains the dashboard. See <code>Folder</code>.</p>
FolderName	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Name of the folder that contains the dashboard. Available in API version 35.0 and later.</p>
IsDeleted	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is <code>Deleted</code>.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
LeftSize	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Returns the size of the left column of the dashboard. Available values are:<ul style="list-style-type: none"> • Narrow • Medium • Wide </p>
MiddleSize	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Returns the size of the middle column of the dashboard. Available values are:</p>

Field	Details
	<ul style="list-style-type: none">• Narrow• Medium• Wide
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none">• In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.• In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
RightSize	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Returns the size of the right column in the dashboard. Available values are:<ul style="list-style-type: none">• Narrow• Medium• Wide</p>
RunningUserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description Returns the ID of the running user specified for the dashboard. If the dashboard is dynamic, it returns the user ID of the viewing user.</p>
TextColor	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the body text color in hexadecimal. Label is <code>Text Color</code>.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the title of the dashboard. Limit: 80 characters.</p>
TitleColor	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the title text color in hexadecimal. Label is <code>Title Color</code>.</p>
TitleSize	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Returns the title font size in points. Label is <code>Title Size</code>.</p>
Type	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Returns the dashboard type. Available values are:</p> <ul style="list-style-type: none"> • <code>SpecificUser</code>—The dashboard displays data according to the access level of one specific running user. • <code>LoggedInUser</code>—The dashboard displays data according to the access level of the logged-in user.

Field	Details
	<ul style="list-style-type: none">MyTeamUser—The dashboard displays data according to the access level of the logged-in user, and managers can view dashboards from the point of view of users beneath them in the role hierarchy.

Supported Query Scopes

Use these scopes to help specify the data that your SOQL query returns.

allPrivate

Records saved in all users' private folders.

Requires the user permission "Manage All Private Reports and Dashboards" and [Enhanced Analytics Folder Sharing](#). If your organization was created after the Summer '13 release, you already have Enhanced Analytics Folder Sharing. Available in API version 36.0 and later.

created

Records created by the user running the query.

everything

All records except records saved in other users' private folders.

mine

Records saved in the private folder of the user running the query.

Usage

Provides read only access to the current values in the dashboard fields.

Example: Dashboards in an Inactive User's Private Folder

This SOQL query returns dashboards saved in a specific user's private folder.

```
SELECT Id FROM Dashboard USING SCOPE allPrivate WHERE CreatedByID = '005A0000000Bc2deFG'
```

SEE ALSO:

[DashboardFeed](#)
[DashboardTag](#)
[Report](#)

DashboardComponent

Represents a dashboard component, which can be a chart, metric, table, or gauge on a dashboard. Access is read-only. This object is available in API version 21.0 and later.

Supported Calls

```
describeSObjects(), query(), retrieve()
```

Fields

Field	Details
CustomReportId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Requires the user permission "Manage All Private Reports and Dashboards." The ID of the report that provides data for the dashboard component. See Report.</p>
DashboardId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the dashboard that contains the dashboard component. See Dashboard.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the dashboard component.</p>

Usage

Provides read only access to the current values in dashboard component fields.

DashboardComponentFeed

Represents a single feed item in the feed displayed on a dashboard component. This object is available in API version 21.0 and later.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of DashboardComponentFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Create, Nillable, Sort</p>

Field	Details
	<p>Description Available in API version 35.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Create, Group, Nillable, Sort</p> <p>Description Available in API version 35.0 and earlier only. This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDate DESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only.</p> <p>ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none">• <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none">• <code></code>• <code><code></code>• <code><i></code>• <code><u></code>• <code><s></code>• <code></code>

Field	Details
	<ul style="list-style-type: none"> • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Create, Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the dashboard that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion object associated with a ContentPost. For Work.com thanks posts, it's the ID of the WorkThanks object associated with a RypplePost. This field is typically null for all posts except ContentPost and RypplePost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name. The Title field can be updated on posts of Type QuestionPost.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description The type of FeedItem. Except for ContentPost, LinkPost, and TextPost, don't create FeedItem types directly from the API.</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL. PollPost—a poll posted on a feed. ProfileSkillPost—generated when a skill is added to a user's Chatter profile. QuestionPost—generated when a user posts a question. ReplyPost—generated when Chatter Answers posts a reply. RipplePost—generated when a user creates a Thanks badge in Work.com. TextPost—a direct text entry on a feed. TrackedChange—a change or group of changes to a tracked field.

Field	Details
	<ul style="list-style-type: none"> UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> AttachArticleEvent—generated event when a user attaches an article to a case. CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. CaseCommentPost—generated event when a user adds a case comment for a case object. ChangeStatusPost—generated event when a user changes the status of a case. ChatTranscriptPost—generated event when Live Agent transcript is saved to a case. EmailMessageEvent—generated event when an email related to a case object is sent or received. FacebookPost—generated when a Facebook post is created from a case. Deprecated. MilestoneEvent—generated when a case milestone is completed or reaches violation status. SocialPost—generated when a social post is created from a case. <p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p>
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none"> AllUsers—The feed item is available to all users who have permission to see the feed item. InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none"> For record posts, Visibility is set to InternalUsers for all internal users by default. External users can set Visibility only to AllUsers. Visibility can be updated on record posts. The Update property is supported only for feed items posted on records.

Usage

Use this object to retrieve the current contents of the feed fields, such as type of feed or feed ID.

DashboardFeed

Represents a single feed item in the feed displayed on a dashboard. This object is available in API version 20.0 and later.

Shows changes to dashboard fields tracked in feeds, and posts and comments about the dashboard.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of DashboardFeed. Required when Type is <code>TextPost</code>. Optional when Type is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type <code>int</code></p> <p>Properties <code>Filter</code>, <code>Group</code>, <code>Sort</code></p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your</p>

Field	Details
	<p>comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p>
	<p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
<code>ConnectionId</code>	Type reference
	Properties Filter, Group, Nillable, Sort
	Description
	<p>When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
<code>ContentData</code>	Type base64
	Properties Nillable
	Description
	<p>Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
<code>ContentDescription</code>	Type textarea
	Properties Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
<code>ContentFileName</code>	Type string
	Properties Group, Nillable, Sort
	Description
	<p>Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>

Field	Details
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.

Field	Details
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the dashboard that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used.

Field	Details
	<ul style="list-style-type: none"> • <code>CanvasPost</code>—a post made by a canvas app posted on a feed. • <code>CollaborationGroupCreated</code>—generated when a user creates a public group. • <code>CollaborationGroupUnarchived</code>—Not used. • <code>ContentPost</code>—a post with an attached file. • <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher. • <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold. • <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed. • <code>LinkPost</code>—a post with an attached URL. • <code>PollPost</code>—a poll posted on a feed. • <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile. • <code>QuestionPost</code>—generated when a user posts a question. • <code>ReplyPost</code>—generated when Chatter Answers posts a reply. • <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Visibility**Type**

picklist

Properties

Filter, Group, Nillable, Restricted picklist, Sort

Description

Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.

`Visibility` can have the following values:

Field	Details
	<ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to retrieve the current contents of the feed fields, such as type of feed or feed ID.

SEE ALSO:

[Dashboard](#)

DashboardTag

Associates a word or short phrase with a Dashboard. This object is available in API version 20.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
<code>ItemId</code>	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
<code>Name</code>	<p>Type string</p> <p>Properties Create, Filter</p>

Field Name	Details
	<p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

DashboardTag stores the relationship between its parent TagDefinition and the Dashboard being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

SEE ALSO:

[Dashboard](#)

DataAssessmentFieldMetric

Represents summary statistics for matched, blank, and differing fields in account records of an org compared to records in Data.com. This object is available in API version 37.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Child Relationships

DataAssessmentFieldMetric is a child object of [DataAssessmentMetric](#) object.

Fields

Field Name	Details
DataAssessmentMetricId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description A unique number that identifies the parent DataAssessmentMetric record.</p>
FieldName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the assessed field.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An optional field used to name your record.</p>
NumMatchedBlanks	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of matched records that contain blank fields.</p>
NumMatchedDifferent	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of matched records that have a different value for this field.</p>

Field Name	Details
NumMatchedInSync	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of matched records that have the same value for this field.</p>
NumUnmatchedBlanks	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of unmatched records that contain blank fields.</p>

DataAssessmentMetric

Represents a summary of statistics for fields matched and unmatched in your account records with Data.com account records. This object is available in API version 37.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An optional field used to name your record.</p>
NumDuplicates	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of duplicate records.</p>

Field Name	Details
NumMatched	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of matched records.</p>
NumMatchedDifferent	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of records in your org matched with a Data.com record that have different fields.</p>
NumProcessed	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of records processed in the data assessment.</p>
NumTotal	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of records available for data assessment processing.</p>
NumUnmatched	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of records not matched.</p>

DataAssessmentValueMetric

Summarizes the number of fields matched for your account records with Data.com account records. This object is available in API version 37.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Child Relationships

DataAssessmentValueMetric is a child of [DataAssessemmentFieldMetric](#).

Fields

Field Name	Details
DataAssessmentFieldMetricId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description A unique number that identifies the parent DataAssessemmentFieldMetric record.</p>
FieldValue	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The value in the matched field.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An optional field used to name your record.</p>
ValueCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of times this value appears in this field.</p>

DatacloudCompany

Represents the fields for Data.com company records. This object is available in API version 30.0 or later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `query()`

Fields

Field Name	Details
ActiveContacts	<p>Type int</p> <p>Properties Nillable</p> <p>Description The number of active contacts that are associated with a company.</p>
AnnualRevenue	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The amount of money that the company makes in 1 year. Annual revenue is measured in US dollars.</p>
City	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The name of the city where the company is located.</p>
CompanyId	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description A unique numerical identifier for the company and theData.com identifier for a company.</p>
Country	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	Description
	A string that represents the standard abbreviation for the country where the company is located.
CountryCode	Type
	picklist
	Properties
	Filter, Group, Nillable, Restricted picklist
	Description
	A standardized name for countries of the world.
Description	Type
	string
	Properties
	Nillable
	Description
	A brief synopsis of the company that provides a general overview of the company and what it does.
DunsNumber	Type
	string
	Properties
	Filter, Nillable
	Description
	A randomly generated nine-digit number that's assigned by Dun & Bradstreet (D&B) to identify unique business establishments.
EmployeeQuantityGrowthRate	Type
	double
	Properties
	Nillable
	Description
	The yearly growth rate of the number of employees in a company expressed as a decimal percentage. The data includes the total employee growth rate for the past two years.
ExternalId	Type
	string
	Properties
	Filter, Nillable, Sort

Field Name	Details
	Description
	<p>A unique numerical identifier for the company. The <code>ExternalId</code> is a system-generated number.</p>
Fax	Type
	phone
	Properties
	Nillable
	Description
	The telephone number that's used to send and receive faxes.
FortuneRank	Type
	int
	Properties
	Defaulted on create, Group, Nillable
	Description
	The numeric value of the company's Fortune 1000 ranking. A null or blank value means that the company isn't ranked as a Fortune 1000 company.
FullAddress	Type
	string
	Properties
	Group, Nillable
	Description
	The complete address of a company, including Street , City , State , and Zip .
IncludedInSnp500	Type
	string
	Properties
	Group, Nillable
	Description
	A true or false value. If <code>true</code> , the company is listed in the S&P 500 Index. If <code>false</code> , the company isn't listed in the S&P 500 Index.
Industry	Type
	string
	Properties
	Nillable
	Description
	A description of the type of industry such as Telecommunications, Agriculture, or Electronics.

Field Name	Details
IsInCrm	Type boolean
	Properties Defaulted on create, Group
	Description Whether the record is in Salesforce (true) or not (false).
IsInactive	Type boolean
	Properties Defaulted on create, Filter
	Description A true or false response. True, the company record is not active. False, the company record is active.
IsOwned	Type boolean
	Properties Defaulted on create
	Description A true or false value. True, your organization owns the record. False, your organization doesn't own the record.
NaicsCode	Type string
	Properties Filter, Nillable
	Description A value that represents the North American Industry Classification System (NAICS) code. NAICS was created to provide details about a business's service orientation. The code descriptions are focused on what a business does.
NaicsDesc	Type string
	Properties Nillable
	Description A description of the NAICS classification.

Field Name	Details
Name	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The company's name.</p>
NumberOfEmployees	<p>Type int</p> <p>Properties Filter, Nillable</p> <p>Description The number of employees working for the company.</p>
Ownership	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The type of ownership of the company:</p> <ul style="list-style-type: none"> • Public • Private • Government • Other
Phone	<p>Type phone</p> <p>Properties Nillable</p> <p>Description A numeric string containing the primary telephone number for the company.</p>
PremisesMeasure	<p>Type int</p> <p>Properties Group, Nillable</p> <p>Description A numeric value for the measurement of the premises.</p>

Field Name	Details
PremisesMeasureReliability	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A descriptive accuracy of the measurement such as actual, estimated, or modeled.</p>
PremisesMeasureUnit	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A descriptive measurement unit such as acres, square meters, or square feet.</p>
PriorYearEmployees	<p>Type int</p> <p>Properties Group, Nillable</p> <p>Description The total number of employees for the prior year.</p>
PriorYearRevenue	<p>Type double</p> <p>Properties Nillable</p> <p>Description The annual revenue for the prior year.</p>
SalesTurnoverGrowthRate	<p>Type double</p> <p>Properties Nillable</p> <p>Description The increase in annual revenue from the previous value for an equivalent period expressed as a decimal percentage.</p>
Sic	<p>Type string</p> <p>Properties Filter, Nillable</p>

Field Name	Details
SicCodeDesc	<p>Description</p> <p>A numeric value that represents the Standard Industrial Codes (SIC). SIC is a numbering convention that indicates what type of service a business provides. It is a four-digit value.</p>
SicDesc	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable</p>
Site	<p>Description</p> <p>The SIC numeric code and descscription for a company.</p>
State	<p>Type</p> <p>string</p> <p>Properties</p> <p>Nillable</p>
StateCode	<p>Description</p> <p>A description of the SIC classification.</p>
Site	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Nillable, Restricted picklist</p>
State	<p>Description</p> <p>An organizational status of the company.</p> <ul style="list-style-type: none"> • Branch: a secondary location to a headquarter location • Headquarter: a parent company with branches or subsidiaries • Single Location: a single business with no subsidiaries or branches
StateCode	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Nillable, Sort</p>
Site	<p>Description</p> <p>The two-letter standard abbreviation for a state.</p>
SiteCode	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Nillable, Restricted picklist</p>

Field Name	Details
Description	
A standard two-letter abbreviation for states and territories of the United States. The state where the company is located. The abbreviation can also be a province or other equivalent to a state, depending on the country where the company is located.	
Street	
Type	
string	
Properties	
Nillable	
Description	
A postal address for the company.	
TickerSymbol	
Type	
string	
Properties	
Nillable	
Description	
The symbol that uniquely identifies companies that are traded on public stock exchanges.	
TradeStyle	
Type	
string	
Properties	
Nillable	
Description	
A legal name under which a company conducts business.	
UpdatedDate	
Type	
dateTime	
Properties	
Nillable, Sort	
Description	
The last date and time when the information for this company was updated.	
Website	
Type	
url	
Properties	
Nillable	

Field Name	Details
	Description
YearStarted	<p>Type string</p> <p>Properties Nillable</p> <p>Description The year when the company was founded.</p>
Zip	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description A numeric postal code that's designated for the address.</p>

Usage

Use the DatacloudCompany object to search the Data.com database for companies with the specific criteria that you enter. Use this object to find company records that you are interested in purchasing for your organization. Data.com APIs use the term “company,” which is similar to Salesforce term “accounts.”

! **Important:** DatacloudCompany can't be used in Apex test methods, because an external web service call is required to access it. These calls are not allowed in Apex test methods.

DatacloudContact

The fields and properties for Data.com contact records. This object is available in API version 30.0 or later.

Supported Calls

`describeSObjects(), query()`

Fields

Field Name	Details
	Type
City	<p>Type string</p>

Field Name	Details
	Properties
	Filter, Nillable, Sort
	Description
	The city where the company is located.
CompanyId	Type
	string
	Properties
	Filter, Nillable
	Description
	The unique numerical identifier for the company and the Data.com company identification number or Data.com Key.
CompanyName	Type
	string
	Properties
	Filter, Group, Sort
	Description
	The name of the company.
ContactId	Type
	string
	Properties
	Filter, Nillable
	Description
	The unique numeric identifier for this contact.
Country	Type
	string
	Properties
	Filter, Nillable, Sort
	Description
	The standard abbreviation or name for the country where the company is located.
	 Note: You can enter a comma-separated list of countries; however, for a country that uses a comma in its name, leave out the comma. For example, enter "Taiwan, ROC" as Taiwan ROC.
Department	Type
	picklist

Field Name	Details
	Properties Filter, Group, Restricted picklist
	Description The department in the company that the contact is affiliated with. The values of this field are fixed enumerated values.
	<ul style="list-style-type: none"> • Engineering • Finance • Human Resources • IT • Marketing • Operations • Other • Sales • Support
Email	Type email
	Properties Filter, Nillable
	Description A business email address for the contact.
ExternalId	Type string
	Properties Filter, Nillable, Sort
	Description A unique system-generated numerical identifier for the contact.
FirstName	Type string
	Properties Filter, Nillable
	Description The first name of the contact.
IsInCrm	Type boolean

Field Name	Details
	Properties
	Defaulted on create, Group
	Description
	Whether the record is in Salesforce (true) or not (false).
IsInactive	Type
	boolean
	Properties
	Defaulted on create, Filter
	Description
	Whether the record is active (false) or not (true).
IsOwned	Type
	boolean
	Properties
	Defaulted on create
	Description
	<ul style="list-style-type: none"> • True: You own this record. • False: You do not own this record.
LastName	Type
	string
	Properties
	Filter, Nillable, Sort
	Description
	The last name of the contact.
Level	Type
	picklist
	Properties
	Filter, Group, Nillable, Restricted picklist
	Description
	A human resource label that designates a person's level in the company. The values of this field are fixed enumerated values.
	<ul style="list-style-type: none"> • C-Level • VP • Director • Manager • Staff • Other

Field Name	Details
Phone	<p>Type phone</p> <p>Properties Nillable</p> <p>Description The direct-dial telephone number for the contact.</p>
SocialHandles	<p>Type string</p> <p>Properties The social handles for this contact. Social handles are a normalized URL and user name for social media accounts such as, LinkedIn, Facebook, and Twitter. This field is response-only.</p> <p>The DatacloudSocialHandles object is a child of the DatacloudContact object.</p>
State	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The state where the company is located, which can also be a province or other equivalent to a state, depending on the country where the company is located.</p>
Street	<p>Type string</p> <p>Properties Nillable</p> <p>Description The street address for the company where the contact works.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Title of the contact such as CEO or Vice President.</p>
UpdatedDate	<p>Type dateTime</p>

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The last date and time when the information for a contact was updated.</p>
zip	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The postal or zip code for the address.</p>

Usage

This object searches the Data.com database for contacts with the specific criteria that you enter. Use this object to find contact records that you are interested in purchasing for your organization.

! **Important:** DatacloudContact can't be used in Apex test methods, because an external web service call is required to access it. These calls are not allowed in Apex test methods.

DatacloudDandBCompany

Represents a set of read-only fields that are used to return D&B company data from Data.com API calls. This object is available in API version 30.0 or later.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field Name	Details
City	<p>Type string</p> <p>Properties Nillable</p> <p>Description The name of the city where the company is physically located.</p>

Field Name	Details
CompanyCurrencyIsoCode	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The code used to represent a company's local currency. This data is provided by the International Organization for Standardization (ISO) and is based on their three-letter currency codes. For example, USD is the ISO code for United States Dollar.</p>
CompanyId	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description A unique numeric identifier for a company.</p>
Country	<p>Type string</p> <p>Properties Nillable</p> <p>Description The country where a company is physically located.</p>
CountryAccessCode	<p>Type string</p> <p>Properties Nillable</p> <p>Description The required code for international calls.</p>
CurrencyCode	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The currency in which the company's sales volume is expressed.</p>
Description	<p>Type string</p>

Field Name	Details
	Properties Nillable
	Description A brief description of the company, which may include information about its history, its products and services, and its influence on a particular industry.
DomesticUltimateBusinessName	Type string
	Properties Nillable
	Description The primary name of the Domestic Ultimate, which is the highest ranking subsidiary, specified by country, within an organization's corporate structure.
DomesticUltimateDunsNumber	Type string
	Properties Nillable
	Description The D-U-N-S number for the Domestic Ultimate, which is the highest-ranking subsidiary, specified by country, within an organization's corporate structure.
DunsNumber	Type string
	Properties Filter, Nillable
	Description The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun & Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used by industries and organizations around the world as a global standard for business identification and tracking.
EmployeeQuantityGrowthRate	Type double
	Properties Nillable
	Description The yearly growth rate of the number of employees in a company expressed as a decimal percentage. The data includes the total employee growth rate for the past two years.

Field Name	Details
EmployeesHere	Type double
	Properties Nillable
	Description The number of employees at a specified location, such as a branch location.
EmployeesHereReliability	Type picklist
	Properties Nillable, Restricted picklist
	Description The reliability of the EmployeesHere figure. Available values are <i>Actual number</i> , <i>Low</i> , <i>Estimated (for all records)</i> , <i>Modeled (for non-US records)</i> . A blank value indicates this data is unavailable.
EmployeesTotal	Type double
	Properties Nillable
	Description The total number of employees in the company, including all subsidiary and branch locations. This data is available only on records that have a value of <i>Headquarters/Parent</i> in the LocationStatus field.
EmployeesTotalReliability	Type picklist
	Properties Nillable, Restricted picklist
	Description The reliability of the EmployeesTotal figure. Available values are <i>Actual number</i> , <i>Low</i> , <i>Estimated (for all records)</i> , <i>Modeled (for non-US records)</i> . A blank value indicates this data is unavailable.
ExternalId	Type string
	Properties Filter, Nillable, Sort
	Description A system generated numeric identification.

Field Name	Details
FamilyMembers	<p>Type int</p> <p>Properties Nillable</p> <p>Description The total number of family members, worldwide, within an organization, including the Global Ultimate, its subsidiaries (if any), and its branches (if any).</p>
Fax	<p>Type phone</p> <p>Properties Nillable</p> <p>Description The company's facsimile number.</p>
FifthNaics	<p>Type string</p> <p>Properties Nillable</p> <p>Description A NAICS code that's used to further classify an organization by industry.</p>
FifthNaicsDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding NAICS code.</p>
FifthSic	<p>Type string</p> <p>Properties Nillable</p> <p>Description A Standard Industrial Classification (SIC) code that's used to further classify an organization by industry.</p>
FifthSic8	<p>Type string</p>

Field Name	Details
	Properties
	Group, Nillable
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
FifthSic8Desc	Type
	string
	Properties
	Group, Nillable
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
FifthSicDesc	Type
	string
	Properties
	Nillable
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code.
FipsMsaCode	Type
	string
	Properties
	Nillable
	Description
	The Federal Information Processing Standards (FIPS) and the Metropolitan Statistical Area (MSA) codes identify the organization's location. The MSA codes are defined by the US Office of Management and Budget.
FipsMsaDesc	Type
	string
	Properties
	Nillable
	Description
	A brief description of an organization's FIPS MSA code.
FortuneRank	Type
	int

Field Name	Details
	<p>Properties</p> <p>Defaulted on create, Group, Nullable</p>
	<p>Description</p> <p>The numeric value of the company's Fortune 1000 ranking. A null or blank value means that the company isn't ranked as a Fortune 1000 company.</p>
FourthNaics	<p>Type</p> <p>string</p>
	<p>Properties</p> <p>Nullable</p>
	<p>Description</p>
	<p>A NAICS code used to further classify an organization by industry.</p>
FourthNaicsDesc	<p>Type</p> <p>string</p>
	<p>Properties</p> <p>Nullable</p>
	<p>Description</p>
	<p>A brief description of an organization's line of business, based on the corresponding NAICS code.</p>
FourthSic	<p>Type</p> <p>string</p>
	<p>Properties</p> <p>Group, Nullable</p>
	<p>Description</p>
	<p>A SIC code used to further classify an organization by industry.</p>
FourthSic8	<p>Type</p> <p>string</p>
	<p>Properties</p> <p>Group, Nullable</p>
	<p>Description</p>
	<p>An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.</p>
FourthSic8Desc	<p>Type</p> <p>string</p>
	<p>Properties</p>
	<p>Group, Nullable</p>

Field Name	Details
	<p>Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>
FourthSicDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding SIC code.</p>
GeoCodeAccuracy	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The level of accuracy of a location's geographical coordinates compared with its physical address. Available values include <i>Rooftop level, Street level, Block level, Census tract level, Mailing address level, ZIP code level, Geocode could not be assigned, Places the address in the correct city, Not matched, Not matched, Street intersection, PO BOX location, and Non-US rooftop accuracy</i>.</p>
GlobalUltimateBusinessName	<p>Type string</p> <p>Properties Nillable</p> <p>Description The primary name of the Global Ultimate, which is the highest entity within an organization's corporate structure and may oversee branches and subsidiaries.</p>
GlobalUltimateDunsNumber	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The D-U-N-S number of the Global Ultimate, which is the highest-ranking entity within an organization's corporate structure and can oversee branches and subsidiaries.</p>

Field Name	Details
GlobalUltimateTotalEmployees	<p>Type double</p> <p>Properties Nillable</p> <p>Description The total number of employees at the Global Ultimate, which is the highest entity within an organization's corporate structure and may oversee branches and subsidiaries.</p>
ImportExportAgent	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description Identifies whether a business imports goods or services, exports goods or services, and/or is an agent for goods.</p>
IncludedInSnP500	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A true or false value. If <code>true</code>, the company is listed in the S&P 500 Index. If <code>false</code>, the company isn't listed in the S&P 500 Index.</p>
Industry	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A description of the type of industry such as Telecommunications, Agriculture, or Electronics.</p>
IsOwned	<p>Type boolean</p> <p>Properties Defaulted on create</p> <p>Description A true or false value. True, your organization owns the record. False, your organization doesn't own the record.</p>

Field Name	Details						
IsParent	<p>Type boolean</p> <p>Properties Defaulted on create,</p> <p>Description A true or false value. True, the company is a parent company. False, the company isn't a parent company. A parent company owns other companies.</p>						
Latitude	<p>Type string</p> <p>Properties Nillable</p> <p>Description Used with longitude to specify a precise location, which is used to assess the Geocode Accuracy.</p>						
LegalStatus	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description Identifies the legal structure of an organization. Available values include <i>Cooperative, Nonprofit organization, Local government body, Partnership of unknown type, and Foreign company.</i></p>						
LocationStatus	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist</p> <p>Description Identifies the organizational status of a company. A numeric value represents each value.</p> <table border="1"> <thead> <tr> <th>Organizational status</th><th>Numeric value</th></tr> </thead> <tbody> <tr> <td><i>Single location:</i> The business has no branches or subsidiaries.</td><td>0</td></tr> <tr> <td><i>Headquarters/Parent:</i> A parent company that owns more than 50 percent of another company. When the company also has branches, it's the headquarters.</td><td>1</td></tr> </tbody> </table>	Organizational status	Numeric value	<i>Single location:</i> The business has no branches or subsidiaries.	0	<i>Headquarters/Parent:</i> A parent company that owns more than 50 percent of another company. When the company also has branches, it's the headquarters.	1
Organizational status	Numeric value						
<i>Single location:</i> The business has no branches or subsidiaries.	0						
<i>Headquarters/Parent:</i> A parent company that owns more than 50 percent of another company. When the company also has branches, it's the headquarters.	1						

Field Name	Details	
	Organizational status	Numeric value
	<i>Branch:</i> A secondary location of a business.	2
 Note: Only the numeric value is accepted in an API request.		
Longitude	Type	string
	Properties	Nillable
	Description	Used with latitude to specify a precise location, which is used to assess the Geocode Accuracy.
MailingCity	Type	string
	Properties	Nillable
	Description	The city where a company has its mail delivered.
MailingCountry	Type	string
	Properties	Nillable
	Description	The country where a company has its mail delivered.
MailingState	Type	string
	Properties	Nillable
	Description	The state where a company has its mail delivered.
MailingStreet	Type	string
	Properties	Nillable

Field Name	Details
	Description
	The street address where a company has its mail delivered.
MailingZip	Type
	string
	Properties
	Nillable
	Description
	The postal zip code for the company.
MarketingPreScreen	Type
	picklist
	Properties
	Nillable, Restricted picklist
	Description
	The probability that a company pays with a significant delay compared to the agreed terms. The risk level is based on the standard Commercial Credit Score, and ranges from low risk to high risk. Available values are <i>High risk of delinquency</i> , <i>Low risk of delinquency</i> , and <i>Moderate risk of delinquency</i> .
	 Important: Use this information for marketing pre-screening purposes only.
MarketingSegmentationCluster	Type
	picklist
	Properties
	Nillable, Restricted picklist
	Description
	Twenty-two distinct, mutually exclusive profiles, created as a result of cluster analysis of Dun & Bradstreet data for US organizations. Available values include <i>High-Tension Branches of Insurance/Utility Industries</i> , <i>Rapid-Growth Large Businesses</i> , <i>Labor-Intensive Giants</i> , <i>Spartans</i> , <i>Main Street USA</i> .
MinorityOwned	Type
	picklist
	Properties
	Nillable, Restricted picklist
	Description
	Indicates whether an organization is owned or controlled by a member of a minority group.

Field Name	Details
Name	Type string Properties Filter, Nillable
	Description The primary or registered name of a company.
NationalId	Type string Properties Nillable
	Description The identification number used in some countries for business registration and tax collection.
NationalIdType	Type picklist Properties Nillable, Restricted picklist
	Description A code value that identifies the type of national identification number that's used.
OutOfBusiness	Type picklist Properties Nillable, Restricted picklist
	Description Indicates whether the company at the specified address has discontinued operations.
OwnOrRent	Type picklist Properties Nillable, Restricted picklist
	Description Indicates whether a company owns or rents the building it occupies.
ParentOrHqBusinessName	Type string

Field Name	Details
	<p>Properties Nillable</p> <p>Description The primary name of the parent or headquarters company.</p>
ParentOrHqDunsNumber	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The D-U-N-S number for the parent or headquarters.</p>
Phone	<p>Type phone</p> <p>Properties Nillable</p> <p>Description A company's primary telephone number.</p>
PremisesMeasure	<p>Type int</p> <p>Properties Group, Nillable</p> <p>Description A numeric value for the measurement of the premises.</p>
PremisesMeasureReliability	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A descriptive accuracy of the measurement such as actual, estimated, or modeled.</p>
PremisesMeasureUnit	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A descriptive measurement unit such as acres, square meters, or square feet.</p>

Field Name	Details
PrimaryNaics	<p>Type string</p> <p>Properties Nillable</p> <p>Description The six-digit North American Industry Classification System (NAICS) code is the standard used by business and government to classify business establishments according to their economic activity for the purpose of collecting, analyzing, and publishing statistical data related to the US business economy.</p>
PrimaryNaicsDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on its NAICS code.</p>
PrimarySic	<p>Type string</p> <p>Properties Nillable</p> <p>Description The four-digit SIC code that's used to categorize business establishments by industry.</p>
PrimarySic8	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The eight-digit Standard Industrial Classification (SIC) code is used to categorize business establishments by industry. The full list of values can be found at the Optimizer Resources page maintained by Dun & Bradstreet. Maximum size is 8 characters.</p>
PrimarySic8Desc	<p>Type string</p> <p>Properties Group, Nillable</p>

Field Name	Details
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
PrimarySicDesc	Type string
	Properties
	Nillable
	Description
	A brief description of an organization's line of business, based on its SIC code.
PriorYearEmployees	Type int
	Properties
	Group, Nillable
	Description
	The total number of employees for the prior year.
PriorYearRevenue	Type double
	Properties
	Nillable
	Description
	The annual revenue for the prior year.
PublicIndicator	Type picklist
	Properties
	Nillable, Restricted picklist
	Description
	Indicates whether ownership of the company is public or private.
Revenue	Type double
	Properties
	Nillable
	Description
	The annual revenue of a company in US dollars.

Field Name	Details
SalesTurnoverGrowthRate	<p>Type double</p> <p>Properties Nillable</p> <p>Description The increase in annual revenue from the previous value for an equivalent period expressed as a decimal percentage.</p>
SalesVolume	<p>Type double</p> <p>Properties Nillable</p> <p>Description The total annual sales revenue in the headquarters' local currency. Dun & Bradstreet tracks revenue data for publicly traded companies, Global Ultimates, Domestic Ultimates, and some headquarters.</p>
SalesVolumeReliability	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The reliability of the SalesVolume figure.</p>
SecondNaics	<p>Type string</p> <p>Properties Nillable</p> <p>Description A NAICS code used to further classify an organization by industry.</p>
SecondNaicsDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding NAICS code.</p>
SecondSic	<p>Type string</p>

Field Name	Details
	<p>Properties Nillable</p> <p>Description A SIC code used to further classify an organization by industry.</p>
SecondSic8	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.</p>
SecondSic8Desc	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>
SecondSicDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding SIC code.</p>
SixthNaics	<p>Type string</p> <p>Properties Nillable</p> <p>Description A NAICS code used to further classify an organization by industry.</p>
SixthNaicsDesc	<p>Type string</p> <p>Properties Nillable</p>

Field Name	Details
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code.
SixthSic	Type
	string
	Properties
	Nillable
	Description
	A SIC code used to further classify an organization by industry.
SixthSic8	Type
	string
	Properties
	Group, Nillable
	Description
	An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.
SixthSic8Desc	Type
	string
	Properties
	Group, Nillable
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.
SixthSicDesc	Type
	string
	Properties
	Nillable
	Description
	A brief description of an organization's line of business, based on the corresponding SIC code.
SmallBusiness	Type
	picklist
	Properties
	Nillable, Restricted picklist

Field Name	Details
	Description
	Indicates whether the company is designated a small business as defined by the Small Business Administration of the US government.
State	Type
	string
	Properties
	Nillable
	Description
	The state where a company is physically located.
StockExchange	Type
	string
	Properties
	Nillable
	Description
	The corresponding exchange for a company's stock symbol, for example, NASDAQ or NYSE.
StockSymbol	Type
	string
	Properties
	Nillable
	Description
	The abbreviation that's used to identify publicly traded shares of a particular stock.
Street	Type
	string
	Properties
	Nillable
	Description
	The street address where a company is physically located.
Subsidiary	Type
	picklist
	Properties
	Nillable, Restricted picklist

Field Name	Details
	<p>Description</p> <p>Indicates whether a company is more than 50 percent owned by another organization.</p>
ThirdNaics	<p>Type string</p> <p>Properties Nillable</p> <p>Description</p> <p>A NAICS code used to further classify an organization by industry.</p>
ThirdNaicsDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description</p> <p>A brief description of an organization's line of business, based on the corresponding NAICS code.</p>
ThirdSic	<p>Type string</p> <p>Properties Nillable</p> <p>Description</p> <p>A SIC code used to further classify an organization by industry.</p>
ThirdSic8	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description</p> <p>An additional SIC code used to further classify an organization by industry. Maximum size is 8 characters.</p>
ThirdSic8Desc	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description</p> <p>A brief description of an organization's line of business, based on the corresponding SIC code. Maximum size is 80 characters.</p>

Field Name	Details
ThirdSicDesc	<p>Type string</p> <p>Properties Nillable</p> <p>Description A brief description of an organization's line of business, based on the corresponding SIC code.</p>
TradeStyle1	<p>Type string</p> <p>Properties Nillable</p> <p>Description A name, different from its legal name, that an organization may use for conducting business. Similar to "Doing business as" or "DBA".</p>
TradeStyle2	<p>Type string</p> <p>Properties Nillable</p> <p>Description A tradestyle used by the organization.</p>
TradeStyle3	<p>Type string</p> <p>Properties Nillable</p> <p>Description A tradestyle used by the organization.</p>
TradeStyle4	<p>Type string</p> <p>Properties Nillable</p> <p>Description A tradestyle used by the organization.</p>
TradeStyle5	<p>Type string</p>

Field Name	Details
	<p>Properties Nillable</p> <p>Description A tradestyle used by the organization.</p>
UsTaxId	<p>Type string</p> <p>Properties Nillable</p> <p>Description The identification number for the company used by the Internal Revenue Service (IRS) in the administration of tax laws. Also referred to as Federal Taxpayer Identification Number.</p>
Website	<p>Type url</p> <p>Properties Filter, Group, Nillable</p> <p>Description An organization's primary website address.</p>
WomenOwned	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description Indicates whether a company is more than 50 percent owned or controlled by a woman.</p>
YearStarted	<p>Type string</p> <p>Properties Nillable</p> <p>Description The year when the company was established or the year when current ownership or management assumed control of the company.</p>
Zip	<p>Type string</p>

Field Name	Details
	<p>Properties Nillable</p>
	<p>Description A five or nine-digit code that's used to help sort mail.</p>

Usage

Use this object to return D&B Company information. These fields are read-only.

! **Important:** DatacloudDandBCompany can't be used in Apex test methods, because an external web service call is required to access it. These calls are not allowed in Apex test methods.

DatacloudOwnedEntity

Represents fields in the DatacloudOwnedEntity object. The DatacloudOwnedEntity object tracks user-purchased records. This object is available in API version 30.0 or later.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
DataDotComKey	<p>Type string</p> <p>Properties Create, Filter, Sort</p> <p>Description The Data.com contact or company record identification number used by the DatacloudPurchaseUsage object to keep track of purchased records. This is equivalent to the Data.com record ID for a contact or company.</p>
DatacloudEntityType	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Sort</p> <p>Description The type of Data.com record you want to purchase.</p>

Field Name	Details
	<ul style="list-style-type: none"> • 0—contact • 1—company
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An optional field used to name your record.</p>
PurchaseType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description A read only field set by the API to identify the purchase type.</p> <ul style="list-style-type: none"> • Added • Export • API
PurchaseUsageId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The unique identification number for the DatacloudPurchaseUsage object created by making a REST POST request.</p> <ul style="list-style-type: none"> • 0—contact • 1—company
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description A unique identifier for the user making the purchase.</p>

Usage

The Datacloud object that tracks records that are purchased and owned by a specific user.

DatacloudPurchaseUsage

Represents an object used to identify and track Data.com record purchases. This object is available in API version 30.0 or later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
DatacloudEntityType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of Data.com record you want to purchase.<ul style="list-style-type: none">• 0—indicates contact entity type.• 1—indicates company entity type.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description An optional field. You can add a description for your purchase.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An optional field used to name your record.</p>
PurchaseType	<p>Type picklist</p>

Field Name	Details
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	A read only field set by the API to identify the purchase type.
	<ul style="list-style-type: none"> ● Added ● Export ● API
Usage	Type
	double
	Properties
	Filter, Sort
	Description
	A read only field set by the API. It is used to track the points used to purchase records.
UserId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	A read only field set by the API that identifies the user purchasing the records.
UserType	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	A read only field set by the API with 2 user types.
	<ul style="list-style-type: none"> ● Monthly Usage ● List Pool User

Usage

The DatacloudPurchaseUsage object allows you to track Data.com record purchases for CRM users.

DatacloudSocialHandle

Returns normalized URLs with userids for different social media used by Data.com contacts. The DatacloudSocialHandle object is a child object of the DatacloudContact object. This object is available in API version 30.0 or later.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field Name	Details
DatacloudContactId	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The unique numeric identifier for a Data.com contact record.</p>
ProviderName	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The name of the social media provider.</p>
SocialId	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The normalized userid for the user on this social media.</p>
Url	<p>Type url</p> <p>Properties Filter, Nullable, Sort</p> <p>Description A normalized URL and userid for the website of the social media provider.</p>

Usage

Returns social handles with Data.com contacts from the DatacloudContact object. Social handle fields are read only fields and can't be used to filter results.

DatedConversionRate

Represents the dated exchange rates used by an organization for which the multicurrency and the effective dated currency features are enabled.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ConversionRate	Type double Properties Filter, Update Description Required. Conversion rate of this currency type against the corporate currency.
IsoCode	Type picklist Properties Filter, Restricted picklist Description Required. ISO code of the currency. Must be one of the valid alphabetic, three-letter currency ISO codes defined by the ISO 4217 standard, such as USD, GBP, or JPY. Must be unique within your organization. Label is Currency ISO Code .
NextStartDate	Type date Properties Filter, Nillable

Field	Details
	<p>Description Read only. The date on which the next effective dated exchange rate will start. Effectively the day after the end date for this exchange rate.</p>
StartDate	<p>Type date</p> <p>Properties Filter</p> <p>Description The date on which the effective dated exchange rate starts.</p>

Usage

This object is for multicurrency organizations with advanced currency management enabled. Use this object to define the exchange rates your organization uses for a date range. This object is not available in single-currency organizations, nor is it available if the organization does not have advanced currency management enabled.

DcSocialProfile

The `DcSocialProfile` object is a read-only object accessible only through the Data.com Social Key API. This object is available in API version 32.0 or later.

Supported Calls

`describeSObjects()`, `query()`

Special Access Rules

This is a read-only object that is accessed by only the Data.com Social Key API

Fields

Field Name	Details
City	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the city where the company is located.</p>

Field Name	Details
CompanyName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The name of the company at which the contact works.</p>
ContactId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The unique numerical identifier for a contact.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A string that represents the standard abbreviation for the country where the contact works.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description An email address for this contact.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name of a contact.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
State	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The two-letter standard abbreviation for a state.</p>

Usage

Provides social handle information to the Data.com Social Key API.

DcSocialProfileHandle

The `DcSocialProfileHandle` object, a child object to `DcSocialProfile`, is a read-only object accessible only through the Data.com Social Key API. The `DcSocialProfileHandle` object is a child object of the `DcSocialProfile` object. This object is available in API version 32.0 or later.

Supported Calls

`describeSObjects()`, `query()`

Special Access Rules

This is a read-only object that is accessed by only the Data.com Social Key API

Fields

Field Name	Details
	Type
<code>DcSocialProfileId</code>	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description A unique identifier for a contacts social handles.</p>

Field Name	Details
Handle	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A normalized social handle for a contact.</p>
ProviderName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the social media provider.</p>

Usage

Provides social handle information to the Data.com Social Key API.

DeclinedEventRelation

Represents invitees with the status Declined for a given event. This object is available in API versions 29.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
EventId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the event.</p>
RelationId	<p>Type reference</p>

Field Name	Details
	Properties
	Filter, Group, Nillable, Sort
	Description
	Indicates the ID of the invitee.
RespondedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	Indicates the most recent date and time when the invitee declined an invitation to the event.
Response	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Indicates the content of the response field. Label is Comment.
Type	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Indicates whether the invitee is a user, lead or contact, or resource.

Usage

Query invitees who have declined an invitation to an event

```
SELECT eventId, type, response FROM DeclinedEventRelation WHERE eventId='00UTD000000ZH5LA'
```

SEE ALSO:

[AcceptedEventRelation](#)

[UndecidedEventRelation](#)

Division

A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.

Supported Calls

```
create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()
```

Special Access Rules

- Divisions must be enabled for your organization to access this object. To discover whether divisions have been enabled for an organization, inspect the User or Group object for the `DefaultDivision` field—if it is present, then divisions have been enabled, and this field (the field is named Division in objects other than User and Group) will be available in all relevant objects.
- Customer Portal users can't access this object.

Fields

Field	Details
<code>IsActive</code>	Type boolean Properties Create, Defaulted on create, Filter, Update Description Indicates whether the division is active (<code>true</code>) or not (<code>false</code>). Label is Active .
<code>IsGlobalDivision</code>	Type boolean Properties Defaulted on createFilter Description Indicates whether the division is your organization's global default division (<code>true</code>) or not (<code>false</code>). Label is Global Division .
<code>Name</code>	Type string Properties Create, Filter, Update Description A descriptive name for the division. Limit: 80 characters.

Field	Details
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The order in which this division name appears in the Division picklist field when creating or editing users in the Salesforce user interface.</p>

Usage

The values available for that field are the global division ID for the organization, created when divisions are first enabled, and any other division IDs that have been created. The division ID associated with a user is populated in the objects owned or created by the user.

You can use the division ID to make searches, reports, and list views run more quickly and return more relevant results if an organization has very large data sets. For more information, see the Salesforce online help, in the Fields description for the object.

You can use WITH in SOSL to pre-filter results based on division. This is faster than specifying the division in a WHERE clause.



Note: The User object has a Division field that is unrelated to this object. The Division field is a standard text field similar to Company or Department that has no special properties. Do not confuse it with the DefaultDivision field, which does relate to this object.

SEE ALSO:

[Object Basics](#)

DivisionLocalization

When the Translation Workbench is enabled for your organization, the DivisionLocalization object provides the translation of the label for a division.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- Your organization must be using Professional, Enterprise, Developer, Unlimited, or Performance Edition and be enabled for the Translation Workbench.
- To view this object, you must have the “View Setup and Configuration” permission.

Fields

Field	Details
Language	<p>Type string</p> <p>Properties Create, Filter, Nillable, Restricted picklist</p> <p>Description The language for this translated label.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix</i>__<i>componentName</i> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Nillable</p> <p>Description The ID of the Division associated with the label that is being translated.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The actual translated label for the division. Label is Translation.</p>

Usage

Use this object to translate the labels of your divisions into the different languages supported by Salesforce.

Document

Represents a file that a user has uploaded. Unlike Attachment records, documents are not attached to a parent object.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()`

Special Access Rules

You must have the “Edit” permission on documents and the appropriate access to the Folder that contains a document in order to create or update a document in that Folder.

Fields

Field	Details
AuthorId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the User who is responsible for the Document.
Body	Type base64 Properties Create, Nillable, Update Description Required. Encoded file data. If specified, then do not specify a URL.
BodyLength	Type int Properties Filter, Group, Sort Description Size of the file (in bytes).

Field	Details
ContentType	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Type of content. Label is Mime Type. Limit: 120 characters. If the Don't allow HTML uploads as attachments or document records security setting is enabled for your organization, you cannot upload files with the following file extensions: .htm, .html, .htt, .htx, .mhtm, .mhtml, .shtm, .shtml, .acgi, .svg.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of the Document. Limit: 255 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Document Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
FolderId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. ID of the Folder that contains the document.</p>
IsBodySearchable	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the contents of the object can be searched using a SOSL FIND call. The ALL FIELDS search group includes the content as a searchable field.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (true) or not (false). Label is Deleted.</p>
IsInternalUseOnly	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the object is only available for internal use (true) or not (false). Label is Internal Use Only.</p>
IsPublic	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the object is available for external use (true) or not (false). Label is Externally Available.</p>
Keywords	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Keywords. Limit: 255 characters.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p>

Field	Details
	Description
	The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date
	Properties
	Filter, Nillable, Sort, Update
	Description
	The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
Name	Type string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Required. Name of the document. Label is Document Name .
NamespacePrefix	Type string
	Properties
	Filter, Group, Sort, Nillable
	Description
	The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.
	The namespace prefix can have one of the following values:
	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Type	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	<p>Description File type of the Document. In general, the values match the file extension for the type of Document (such as pdf or jpg). Label is File Extension.</p>
Url	<p>Type string</p> <p>Properties Create, Filter, Nillable, Group, Sort, Update</p> <p>Description URL reference to the file (instead of storing it in the database). If specified, do not specify the Body or BodyLength.</p>

Usage

When creating or updating a document, you can specify a value in either the Body or Url fields, but not both.

Encoded Data

The API sends and receives the binary file data encoded as a base64 data type. Prior to creating a record, clients must encode the binary file data as base64. Upon receiving an API response, clients must decode the base64 data to binary (this conversion is usually handled for you by the SOAP client).

Maximum Document Size

You can only create or update documents to a maximum size of 5 MB.

SEE ALSO:

[Object Basics](#)

DocumentAttachmentMap

Maps the relationship between an EmailTemplate and its attachment, which is stored as a Document.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
DocumentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the document that this object tracks.</p>
DocumentSequence	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Represents the order that the attachments will be included in the email defined by the EmailTemplate specified by the DocumentId. Label is Attachment Sequence. The first attachment is given a value of 0, and each subsequent attachment is given a value incremented by 1.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the EmailTemplate parent. The attachment identified by DocumentId is attached to the EmailTemplate specified in this field.</p>

Usage

Use this object to map the relationship of an EmailTemplate to its attachments, and to specify the order of the attachments.

SEE ALSO:

[EmailTemplate](#)

DocumentTag

Associates a word or short phrase with a Document.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

DocumentTag stores the relationship between its parent TagDefinition and the Document being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

Domain

Read-only object that represents a custom Web address assigned to a site in your organization. This object is available in API version 26.0 and later.

To access this object, Force.com Sites or Site.com must be enabled for your organization.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

- Customer Portal users can't access this object.
- To view this object, you must have the "View Setup and Configuration" permission.
- Site.com Publisher users have read-only API access to the Domain and [DomainSite](#) objects.

Fields

Field	Description
Domain	<p>Type string</p> <p>Properties Filter, Sort</p> <p>Description The branded custom Web address within the global namespace identified by this domain's type. In the Domain Name System (DNS) global namespace, this field is the custom Web address that you registered with a third-party domain name registrar. The custom Web address can be used to access the site of this domain.</p>
DomainType	<p>Type picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description The global namespace that this custom Web address belongs to. This value is set to DNS for custom Web addresses in the global DNS.</p> <p>DomainType can have the following values:</p> <ul style="list-style-type: none">• Domain Name System (DNS)

Field	Description
	<ul style="list-style-type: none"> Facebook Page
OptionsExternalHttps	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the domain supports secure connections (true) or not (false) via a non-Salesforce content delivery network (CDN) or endpoint. The value of this field is used only if the domain is not pointing to the <code>yourdomain.your18characterOrgId.live.siteforce.com</code> CNAME target.</p>

Usage

Use this read-only object to query the domains that are associated with each website in your organization.

DomainSite

Read-only junction object that joins together the Site and Domain objects. This object is available in API version 26.0 and later.

To access this object, Force.com Sites or Site.com must be enabled for your organization.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

- Customer Portal users can't access this object.
- To view this object, you must have the "View Setup and Configuration" permission.
- Site.com Publisher users have read-only API access to the [Domain](#) and [DomainSite](#) objects.

Fields

Field	Description
DomainId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field	Description
	Description
PathPrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Shows where a site's root exists on a domain. Can only be set for custom Web addresses. Always begins with a /.</p>
SiteId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the associated Site.</p>

Usage

Use this read-only object to query or retrieve information about your Force.com site.

DuplicateRecordItem

Represents an individual record that's part of a duplicate record set. Use this object to create custom report types.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

To access this object, your organization must have Data.com Duplicate Management enabled. Data.com Duplicate Management is currently available as a beta feature. Contact Salesforce for more information. For users to access this object, they must be given read and write access by their system administrator.

Fields

Field Name	Details
DuplicateRecordSetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The duplicate record set that the duplicate record item is assigned to.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The autogenerated name that's given to the Duplicate Record Item. Label is Duplicate Record Item Name.</p>
RecordId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the record as it appears on the record's detail page.</p>

DuplicateRecordSet

Represents a group of records that have been identified as duplicates. Each duplicate record set contains one or more duplicate record items. Use this object to create custom report types.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

To access this object, your organization must have Data.com Duplicate Management enabled. Data.com Duplicate Management is currently available as a beta feature. Contact Salesforce for more information. For users to access this object, they must be given read and write access by their system administrator.

Fields

Field Name	Details
DuplicateRuleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The duplicate rule that's used to identify this list of duplicate records.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The autogenerated name that's given to the duplicate record set. Label is <code>Duplicate Record Set Name</code>.</p>
RecordCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of record items in the set.</p>

Field Name	Details
SObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The object that the duplicate rule and all the record items in the set apply to.</p>

DuplicateRule

Represents a duplicate rule for detecting duplicate records.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `search()`

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The developer name for the duplicate rule.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a duplicate rule is active (<code>true</code>) or not (<code>false</code>). This field is read only.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language for the duplicate rule.</p>

Field Name	Details
MasterLabel	Type string
	Properties Filter, Group, Sort
	Description The label for the duplicate rule.
NamespacePrefix	Type string
	Properties Filter, Group, Nillable, Sort
	Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.
	The namespace prefix can have one of the following values:
	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
SobjectType	Type picklist
	Properties Filter, Group, Restricted picklist, Sort
	Description The type of object the duplicate rule is defined for. For example, account, contact, or lead.

Usage

DuplicateRule is available to organizations that use duplicate rules.

You can only view details of a duplicate rule using the API. You must use the full Salesforce site to create, edit, and delete duplicate rules.

EmailDomainKey

Represents a domain key for an organization's domain, used to authenticate outbound email that Salesforce sends on the organization's behalf. This object is available in API version 28.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
Domain	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The organization's domain name that the DKIM key is generated for.</p>
DomainMatch	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The specificity of match required on the sending domain name before signing with this DKIM key. Valid values are:</p> <ul style="list-style-type: none">• DomainOnly—Sign if sending domain matches at the domain level only (example.com but not mail.example.com)• SubdomainsOnly—Sign if sending domain matches at the subdomain level only (mail.example.com but not example.com)• DomainAndSubdomains—Sign if sending domain matches at the domain and subdomain levels (example.com and mail.example.com)
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this DKIM key is active (<code>true</code>) or not (<code>false</code>). You can set <code>IsActive</code> to <code>true</code> on create if you specify an existing key pair.</p>

Field Name	Details
PrivateKey	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The private portion of the DKIM key pair used to encrypt mail headers from your domain. Salesforce generates an encrypted <code>PrivateKey</code> if you don't specify a value when creating the DKIM key. If you do specify a value, it must be an existing valid <code>PrivateKey</code> from another <code>EmailDomainKey</code> object.</p> <p>This field doesn't contain the actual private key, but a value that represents the key in our system. Therefore:</p> <ul style="list-style-type: none"> • The actual private key can't be leaked. • You can't use the value to do your own email signing.
PublicKey	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Part of the domain key pair that mail recipients retrieve to decrypt the DKIM header and verify your domain. Add the <code>PublicKey</code> value to your domain's DNS records before you start signing with this domain key. Otherwise, mail recipients may reject your email.</p>
Selector	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Text used to distinguish the DKIM key from any other DKIM keys your organization uses for the specified domain.</p>

Usage

Use this object to define a DomainKeys Identified Mail (DKIM) key, which is used to authenticate outbound email that Salesforce sends on your organization's behalf. When you create a DKIM key, Salesforce generates a public and private key pair. You must publish the public key in the DNS, which tells recipients that you, as the owner of the domain, have authorized the use of this key to sign your mail. Salesforce uses the private key to create DKIM signature headers on your outgoing email. Then, recipients of the mail, can compare the signature header with the public key in the DNS to determine that the mail was signed with an authorized key. If your domain also publishes a Domain-based Message Authentication, Reporting and Conformance (DMARC) policy, recipients can use the DKIM signature to verify that the mail conforms to DMARC.

For each domain key you create, we recommend this sequence:

1. Insert the Domain, DomainMatch, and Selector.
2. Update your domain's DNS records.
 - a. Locate the DNS record at **selector._domainkey.domain**. For example, mail._domainkey.mail.example.com.
 - b. Add the PublicKey value, like this: v=DKIM1; p=**public_key**.
- DKIM Signing Outbound Email
 - a. In addition, you can optionally put the record in testing mode, which instructs recipients to not make decisions based on the email signature. Add parameter t=y to the DNS entry, like this: v=DKIM1; t=y; p=**public_key**.

3. Update the key via the API or UI to be active.

Consider the following when using domain keys.

- Make sure you add the public key to your DNS record before you make your key active in Salesforce and start DKIM signing. DKIM signing is active whenever your DKIM key is in the active state.
- You can't have more than one active DKIM key per domain name. You might have multiple active DKIM keys if your organization mails from more than a single domain or if you use subdomains under your organizational domain and have specified domain matching at the subdomain level.
- If you want to use the same DKIM key for multiple organizations, you can. Create the key and ensure it's working for one organization first. Then using the API or UI create the key in your other organizations by setting the corresponding fields in the new key to the same values as the original.
- When you insert or update a DKIM key, it's possible that the change affects existing domain keys. For example, if you've set DomainMatch to DomainAndSubdomains for the example.com domain, and you then set DomainMatch to SubdomainsOnly for the mail.example.com domain, either key could be used. Here's how we resolve conflicts in the case when DKIM keys overlap.
 - If two keys are equally specific about matching for the same domain, the new key replaces and deactivates the existing key.
 - If a new key is more specific about matching than an existing key, the new key is used and the existing key is modified to inactive.
 - If multiple keys have different domains that match the sending domain, the key with the longest domain name is used. In case of a tie, the most specific key is used. For example, because DomainOnly and SubdomainsOnly are more specific than DomainAndSubdomains, a new DomainOnly key would change the DomainMatch for an existing DomainAndSubdomains key to become SubdomainsOnly. In case of a tie, the most specific key is used.

EmailMessage

Represents an email in Salesforce.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

EmailMessage is only available for organizations that use Email-to-Case or Enhanced Email, which is automatically enabled for most customers.

Customer Portal users have read access to EmailMessage if the value for the `ParentID` field is associated with a case. Otherwise, access is determined by sharing access of the associated task.

`update()` is supported only on records whose `Status` is `Draft`. When a record's `Status` is *not* `Draft`, `update()` is supported only for the `IsExternallyVisible` field and custom fields.

Fields

Field	Details
<code>ActivityId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the activity that is associated with the email. Usually represents an open task that is created for the case owner when a new unread email message is received.</p>
<code>BccAddress</code>	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A string array of email addresses for recipients who were sent a blind carbon copy of the email message. This field should include only email addresses that are not associated with Contact, Lead, or User records in Salesforce. If the recipient is a contact, lead, or user, add their ID to the <code>BccIds</code> field instead of adding their email address to the <code>BccAddress</code> field. Then the email message is automatically associated with the contact, lead, or user.</p>
<code>BccIds</code>	<p>Type <code>JunctionIdList</code></p> <p>Properties Create, Update</p> <p>Description A string array of IDs for contacts, leads, and users who were sent a blind carbon copy of the email message. Each ID is linked to an <code>EmailMessageRelation</code> record, which represents the relationship between an email message and a Contact, Lead, or User record.</p> <p> Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.</p>
<code>CcAddress</code>	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description A string array of email addresses for recipients who were sent a carbon copy of the email message. This field should include only email addresses that are not associated with Contact, Lead, or User records in Salesforce. If the recipient is a contact, lead, or user, add their ID to the <code>CcIds</code> field instead of adding their email address to the <code>CcAddress</code> field. Then the email message is automatically associated with the contact, lead, or user.</p>
CcIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of IDs for contacts, leads, and users who were sent a carbon copy of the email message. Each ID is linked to an <code>EmailMessageRelation</code> record, which represents the relationship between an email message and a Contact, Lead, or User record.</p> <p> Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.</p>
ContentDocumentIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of IDs for content documents, such as files and attachments, that are associated with an email. Each ID is linked to a <code>ContentDocumentLink</code> record, which represents the relationship between an email message and a content document record.</p> <p> Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.</p>
FromAddress	<p>Type email</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The address that originated the email.</p>

Field	Details
FromName	<p>Type string</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The sender's name.</p>
HasAttachment	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the email was sent with an attachment (<code>true</code>) or not (<code>false</code>).</p>
Headers	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description The Internet message headers of the incoming email. Used for debugging and tracing purposes. Does not apply to outgoing emails.</p>
HtmlBody	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description The body of the email in HTML format.</p>
Incoming	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter</p> <p>Description Indicates whether the email was received (<code>true</code>) or sent (<code>false</code>).</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p>

Field	Details
	<p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsExternallyVisible	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Update</p> <p>Description If the community case feed is enabled, <code>IsExternallyVisible</code> controls the external visibility of emails in communities. When <code>IsExternallyVisible</code> is set to <code>true</code>—its default value—email messages are visible to external users in the case feed.</p> <p>Only emails with the <code>ParentId</code> field populated are available to be externally visible in communities.</p> <p>This field can't be updated if the email's <code>Status</code> is set to <code>Draft</code>.</p>
MessageDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The date the email was created.</p>
ParentId	<p>Type ID</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the case that's associated with the email.</p>
RelatedToId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The <code>RelatedToId</code> represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. RelatedToids are polymorphic. Polymorphic means a <code>RelatedToid</code> is equivalent to the ID of a related object.</p>
ReplyToEmailMessageId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the inbound or outbound EmailMessage the current EmailMessage is a reply to. It's not possible to reply to a message whose <code>Status</code> is <code>Draft</code>.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description Read only. The status of the email. For example, <code>New</code>, <code>Draft</code>, <code>Unread</code>, <code>Replied</code>, or <code>Sent</code>.</p>
Subject	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The subject line of the email.</p>
TextBody	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The body of the email, in plain text format.</p>
ToAddress	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description A string array of email addresses for recipients who were sent the email message. This field should include only email addresses that are not associated with Contact, Lead, or User records in Salesforce. If the recipient is a contact, lead, or user, add their ID to the <code>ToIds</code> field instead of adding their email address to the <code>ToAddress</code> field. Then the email message is automatically associated with the contact, lead, or user.</p>
ToIds	<p>Type JunctionIdList</p>

Field	Details
Properties	Create, Update
Description	A string array of IDs for contacts, leads, and users who were sent the email message. Each ID is linked to an <code>EmailMessageRelation</code> record, which represents the relationship between an email message and a Contact, Lead, or User record.

 **Warning:** Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.

Usage

If your organization uses Email-to-Case, a case is created when an email is sent to one of your company's addresses. The email, which is related to the case by the `ParentID` field, is stored as an `EmailMessage` record. When users view the email, they see the `EmailMessage` record.

If your organization uses Enhanced Email, each email is stored as an `EmailMessage` record and a Task record. When users view an email, they see the `EmailMessage` record.

SEE ALSO:

[Case](#)
[Object Basics](#)

EmailMessageRelation

Represents the relationship between an email and contacts, leads, and users. This object is available in API version 37.0 and later

Special Access Rules

`EmailMessageRelation` is only available for organizations that use Email-to-Case or Enhanced Email, which is automatically enabled for most customers.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`update()`,`upsert()`

Fields

Field Name	Details
<code>EmailMessageId</code>	
	Type reference

Field Name	Details
	<p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the EmailMessage record.</p>
RelationAddress	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The email address of the sender or recipient.</p> <p> Note: If a record relates an email to an existing contact, lead, or user record in Salesforce, the value of RelationAddress is null.</p>
RelationId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The RecordId of the sender or recipient.</p> <p> Note: If a record relates an email to an email address that's not associated with an existing contact, lead, or user record in Salesforce, the value of RelationId is null.</p>
RelationObjectType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The API name of the object type of the RecordId in the RelationId field. It can be a contact, lead, or user.</p>
RelationType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The type of relationship the contact, lead, or user has with the email message. Possible values include:</p> <ul style="list-style-type: none"> • ToAddress • CcAddress

Field Name	Details
	<ul style="list-style-type: none"> BccAddress FromAddress

Usage

EmailMessageRelation allows an email to be related to contacts, leads, and users.

EmailServicesAddress

An email service address.

Each email service has one or more email addresses to which users can send messages for processing. An email service only processes messages it receives at one of its addresses.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
AuthorizedSenders	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Configures the email service address to only accept messages from the email addresses or domains listed in this field. If the email service address receives a message from an unlisted email address or domain, the email service performs the action specified in the AuthorizationFailureAction field of its associated email service. Leave this field blank if you want the email service address to receive email from any email address.</p>
EmailDomainName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A read only field you can query that contains the system-generated domain part of this email service address. The system generates a unique domain-part for each email service address to ensure that no two email service addresses are identical.</p>

Field	Details
FunctionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the email service for which the email service address receives messages.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this object is active (true) or not (false).</p>
LocalPart	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The local-part of the email service address. For the local-part of a Salesforce email address, all alphanumeric characters are valid, plus the following special characters: ! # \$ % & &#38;lt;br&gt; ! * / = ? ^ _ + - ` { } ~ , The dot character (.) is also valid as long as it's not the first or last character. Email addresses aren't case sensitive.</p>
RunAsUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The user whose permissions the email service assumes when processing messages sent to this address.</p>

Usage

This object supports the email services feature, which allows you to create automated processes that use Apex classes to process the contents, headers, and attachments of inbound email. For example, you can create an email service that automatically creates contact records based on contact information in messages.

SEE ALSO:

[EmailServicesFunction](#)

EmailServicesFunction

An email service.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
AddressInactiveAction	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates what the email service does with messages received at an email address that is inactive. One of the following values:</p> <ul style="list-style-type: none">• 0—The system default is used.• 1—The email service returns the message to the sender with a notification that explains why the message was rejected.• 2—The email service deletes the message without notifying the sender.• 3—The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.
ApexClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description Required. The ID of the Apex class that the email service uses to process inbound messages. This field is required for API version 12.0 and later.</p>
AttachmentOption	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Indicates the types of attachments the email service accepts. One of the following values:</p> <ul style="list-style-type: none"> • 0—The email service accepts the message but discards any attachment. • 1—The email service only accepts the following types of attachments: <ul style="list-style-type: none"> — Attachments with a Multipurpose Internet Mail Extension (MIME) type of text. — Attachments with a MIME type of application/octet-stream and a file name that ends with either a .vcf or .vcs extension. These are saved as text/x-vcard and text/calendar MIME types, respectively. • 2—The email service only accepts binary attachments, such as image, audio, application, and video files. • 3—The email service accepts any type of attachment.
AuthenticationFailureAction	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates what the email service does with messages that fail or do not support any of the authentication protocols if the <code>IsAuthenticationRequired</code> field is true.</p> <p>One of the following values:</p> <ul style="list-style-type: none"> • 0—The system default is used. • 1—The email service returns the message to the sender with a notification that explains why the message was rejected. • 2—The email service deletes the message without notifying the sender. • 3—The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.
AuthorizationFailureAction	<p>Type picklist</p> <p>Properties Defaulted on create, Group, Sort, Create, Filter, Nillable, Restricted picklist, Update</p>

Field	Details
	<p>Description</p> <p>Indicates what the email service does with messages received from senders who are not listed in the <code>AuthorizedSenders</code> field on either the email service or email service address.</p> <p>One of the following values:</p> <ul style="list-style-type: none"> • 0—The system default is used. • 1—The email service returns the message to the sender with a notification that explains why the message was rejected. • 2—The email service deletes the message without notifying the sender. • 3—The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.
<code>AuthorizedSenders</code>	<p>Type</p> <p>textarea</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>Configures the email service to only accept messages from the email addresses or domains listed in this field. If the email service receives a message from an unlisted email address or domain, the email service performs the action specified in the <code>AuthorizationFailureAction</code> field. Leave this field blank if you want the email service to receive email from any email address.</p>
<code>ErrorRoutingAddress</code>	<p>Type</p> <p>email</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The destination email address for error notification email messages when <code>IsErrorRoutingEnabled</code> is <code>true</code>.</p>
<code>FunctionInactiveAction</code>	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description</p> <p>Indicates what the email service does with messages it receives when the email service itself is inactive.</p> <p>One of the following values:</p> <ul style="list-style-type: none"> • 0—The system default is used.

Field	Details
	<ul style="list-style-type: none"> 1—The email service returns the message to the sender with a notification that explains why the message was rejected. 2—The email service deletes the message without notifying the sender. 3—The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.
FunctionName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the email service.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this object is active (<code>true</code>) or not (<code>false</code>).</p>
IsAuthenticationRequired	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Configures the email service to verify the legitimacy of the sending server before processing a message. The email service uses the SPF, SenderId, and DomainKeys protocols to verify the sender's legitimacy: If the sending server passes at least one of these protocols and does not fail any, the email service accepts the email. If the server fails a protocol or does not support any of the protocols, the email service performs the action specified in the <code>AuthenticationFailureAction</code> field.</p>
IsErrorRoutingEnabled	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description When incoming email messages can't be processed, indicates whether error notification email messages are routed to a chosen address or to the senders.</p>
IsTextAttachmentsAsBinary	<p>Type boolean</p>

Field	Details
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates whether the email service converts text attachments to binary files.
IsTextTruncated	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	This field is deprecated. It is not available as of API version 23.0 and is deprecated and hidden in versions 17.0 through 22.0. In all API versions, the email service now accepts inbound email messages up to the 10 MB size limit, without truncating the text. Previously, it indicated whether the email service truncated and accepted email messages with HTML body text, plain body text, and text attachments over approximately 100,000 characters (<code>true</code>) or rejected these email messages and notified the sender (<code>false</code>).
IsTlsRequired	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Configures the email service to use Transport Layer Security (TLS), a protocol for secure email communication, to ensure the security and authenticity of inbound email.
OverLimitAction	Type
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	Indicates what the email service does with messages if the total number of messages processed by all email services combined has reached the daily limit for your organization.
	One of the following values:
	<ul style="list-style-type: none"> • 0—The system default is used. • 1—The email service returns the message to the sender with a notification that explains why the message was rejected. • 2—The email service deletes the message without notifying the sender. • 3—The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.
	The system calculates the limit by multiplying the number of user licenses by 1,000.

Usage

This object supports the email services feature, which allows you to create automated processes that use Apex classes to process the contents, headers, and attachments of inbound email. For example, you can create an email service that automatically creates contact records based on contact information in messages.

SEE ALSO:

[EmailServicesAddress](#)

EmailStatus

Represents the status of email sent.

Supported Calls

`describeSObjects()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
EmailTemplateName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the EmailTemplate.</p>
FirstOpenDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date when the email was first opened by recipient. Label is Date Opened.</p>
LastOpenDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description Date when the email was last opened by recipient.</p>
TaskId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The activity (task or event) associated with the email. Label is Activity ID.</p>
TimesOpened	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of times the recipient opened the email.</p>
Whold	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The Whold represents a human such as a lead or a contact. Wholds are polymorphic. Polymorphic means a Whold is equivalent to a contact's ID or a lead's ID. The label is Name ID.</p>

SEE ALSO:

[EmailTemplate](#)

EmailTemplate

Represents a template for mass email, or email sent when the activity history related list of a record is modified.

 **Note:** You can't send a mass email using a Visualforce email template.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ApiVersion	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The API version for this class. Every class has an API version specified at creation.</p>
Body	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Content of the email. Limit: 384 KB.</p>
BrandTemplateId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the BrandTemplate associated with this email template. The brand template supplies letterhead information for the email template.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the template, for example, Promotion Mass Mailing.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not</p>

Field	Details
	<p>include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Template Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
Encoding	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Character set encoding for the template.</p>
EntityType	<p>Type picklist</p> <p>Properties Create, Filter, Group, , Restricted picklist, Sort,</p> <p>Description When <code>UIType</code> is 2 (Lightning Experience) or 3 (Lightning ExperienceSample), <code>EntityType</code> indicates which entities this template can be used with (for example, an account, lead, and so on). Valid values are standard object ID prefixes: 001 for account, 003 for contact, 006 for opportunity, and 00Q for lead, 500 for case, and 701 for campaign.</p>
FolderId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the folder that contains the template.</p>
HtmlValue	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description This field contains the content of the email message, including HTML coding to render the email message. Limit: 384 KB.</p>
IsActive	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates that this template is active if <code>true</code>, or inactive if <code>false</code>.</p>
LastUsedDate	<p>Type <code>dateTime</code></p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when this EmailTemplate was last used.</p>
Markup	<p>Type <code>textarea</code></p> <p>Properties Create, Nillable, Update</p> <p>Description The Visualforce markup, HTML, Javascript, or any other Web-enabled code that defines the content of the template.</p>
Name	<p>Type <code>string</code></p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the template. Label is Email Template Name.</p>
NamespacePrefix	<p>Type <code>string</code></p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values:<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. </p>

Field	Details
	<ul style="list-style-type: none"> In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field cannot be accessed unless the logged-in user has the "Customize Application" permission.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the template.</p>
Subject	<p>Type string</p> <p>Properties Create, Group, Nillable, Sort, Update</p> <p>Description Content of the subject line.</p>
TemplateStyle	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Style of the template, such as formalLetter or freeform.</p>
TemplateType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Type of template, either HTML, text, or custom templates, or those generated by Visualforce.</p>
TimesUsed	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number of times this template has been used.</p>

Field	Details
UIType	<p>Type picklist</p> <p>Properties Create, Filter, Group, , Restricted picklist, Sort,</p> <p>Description Indicates the user interface where this template is usable. Valid values are: 1 (Salesforce Classic), 2 (Lightning Experience), and 3 (Lightning Experience Sample).</p>

Usage

To retrieve this object, issue a describe call on an object, which will return a query result for each activity since the object was created. You can't query these records.

SEE ALSO:

[Attachment](#)
[EmailStatus](#)
[DocumentAttachmentMap](#)

Entitlement

Represents the customer support an Account or Contact is eligible to receive. This object is available in API version 18.0 and later. Entitlements may be based on an Asset, Product2, or ServiceContract.

Supported Calls

`create(), delete(), describeLayout(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the Account associated with the entitlement.</p>
AssetId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nullable, Sort</p>
	<p>Description Required. ID of the Asset associated with the entitlement. Must be a valid asset ID.</p>
BusinessHoursId	<p>Type reference</p>
	<p>Properties Filter, Group, Nullable, Sort</p>
	<p>Description Required. ID of the BusinessHours associated with the entitlement. Must be a valid business hours ID.</p>
CasesPerEntitlement	<p>Type int</p>
	<p>Properties Filter, Group, Nullable, Sort</p>
	<p>Description The total number of cases the entitlement supports. This field is only available if <code>IsPerIncident</code> is true.</p>
ContractLineItemId	<p>Type reference</p>
	<p>Properties Filter, Group, Nullable, Sort</p>
	<p>Description Required. ID of the ContractLineItem associated with the entitlement. Must be a valid ID.</p>
EndDate	<p>Type date</p>
	<p>Properties Create, Filter, Nullable, Update</p>
	<p>Description The last day the entitlement is in effect.</p>
IsPerIncident	<p>Type boolean</p>
	<p>Properties Defaulted on create, Filter, Update</p>

Field	Details
	<p>Description Indicates whether the entitlement is limited to supporting a specific number of cases (<code>true</code>) or not (<code>false</code>).</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Update</p> <p>Description Required. Name of the entitlement.</p>
SvcApptBookingWindowsId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Nillable, Update</p> <p>Description The operating hours that the entitlement's work orders should respect. The label in the user interface is <code>Service Appointment Booking Windows</code>. Available only if Field Service Lightning is enabled.</p>
RemainingCases	<p>Type int</p> <p>Properties Create, Filter, Nillable, Update</p>

Field	Details
	<p>Description The number of cases the entitlement can support. This field decreases in value by one each time a case is created with the entitlement. This field is only available if <code>IsPerIncident</code> is selected.</p>
ServiceContractId	<p>Type reference</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Required. ID of the ServiceContract associated with the entitlement. Must be a valid ID.</p>
SlaProcessId	<p>Type reference</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description ID of the SlaProcess associated with the entitlement. This field is available in version 19.0 and later.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The first date the entitlement is in effect.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Nullable</p> <p>Description Status of the entitlement, such as <code>Expired</code>.</p>
SvcApptBookingWindows	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Nullable, Update</p> <p>Description The operating hours of the entitlement. This field is visible only if Field Service Lightning is enabled.</p>

Field	Details
Type	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Update</p> <p>Description The type of entitlement, such as Web or phone support.</p>

Usage

Use this object to query and manage entitlements.

SEE ALSO:

[EntitlementContact](#)
[EntitlementHistory](#)
[SlaProcess](#)

EntitlementContact

Represents a Contact eligible to receive customer support via an Entitlement. This object is available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete()`

Fields

Field	Details
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Contact associated with the entitlement. Must be a valid ID.</p>
EntitlementId	<p>Type reference</p> <p>Properties Create, Filter</p>

Field	Details
	Description Required. ID of the Entitlement associated with the entitlement contact. Must be a valid ID.
IsDeleted	Type boolean
	Properties Defaulted on create, Filter
	Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
Name	Type string
	Properties Autonumber, Defaulted on create, Filter, idLookup, Nillable
	Description Required. Name of the entitlement contact.

Usage

Use to query and manage entitlement contacts.

SEE ALSO:

[Entitlement](#)
[EntitlementHistory](#)

EntitlementFeed

Represents a single feed item in the feed displayed on the detail page for an entitlement. An entitlement feed shows recent changes to an entitlement record for any fields that are tracked in feeds, and comments and posts about the record. It's a useful way to stay up-to-date with changes to entitlements. This object is available in API version 23.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"

- “Modify All” on the Entitlement object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the EntitlementFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with “CanApproveFeedPostAndComment” or “ModifyAllData” permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The</p>

Field Name	Details
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if Type is ContentPost. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description The ID of the user who added this object to the feed. For example, if a client application migrates multiple posts and comments from another application into a feed, <code>InsertedById</code> is set to the ID of the logged-in user.</p>
IsRichText	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p class="list-item-l1">  Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines. </p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field Name	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on the feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for the FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of the <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default <code>community</code>, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p>

Field Name	Details
	<ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	Type reference
	Properties Filter, Group, Sort
	Description
	ID of the entitlement record that is tracked in the feed. The feed is displayed on the detail page for this record.
RelatedRecordId	Type reference
	Properties Group, Nillable, Sort
	Description
	ID of the ContentVersion object associated with a <code>ContentPost</code> . This field is null for all posts except <code>ContentPost</code> . For example, set this field to an existing ContentVersion and post it to a feed as an EntitlementFeed object of Type <code>ContentPost</code> .
Title	Type string
	Properties Group, Nillable, Sort
	Description
	The title of the EntitlementFeed. When the <code>Type</code> is <code>LinkPost</code> , the <code>LinkUrl</code> is the URL and this field is the link name.
Type	Type picklist
	Properties Filter, Group, Nillable, Restricted picklist, Sort

Field Name	Details
	<p>Description</p> <p>The type of EntitlementFeed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p>

Field Name	Details
	<ul style="list-style-type: none"> • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. • <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case.
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • <code>Visibility</code> can be updated on record posts. • The <code>Update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for an entitlement record.

SEE ALSO:

[Entitlement](#)
[EntitlementHistory](#)
[EntitlementContact](#)
[FeedItem](#)

EntitlementHistory

Represents the changes to field values on an Entitlement. This object is available in API version 18.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
EntitlementId	<p>Type reference</p> <p>Properties Filter</p> <p>Description Required. ID of the Entitlement.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>

Field	Details
NewValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description The latest value of the field before it was changed.</p>

Usage

This object respects field-level security on the parent object.

SEE ALSO:

[Entitlement](#)
[EntitlementContact](#)

EntitlementTemplate

Represents predefined terms of customer support for a product (Product2). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

All users, even those without the “View Setup and Configuration” user permission, can view entitlement templates via the API.

Fields

Field	Details
BusinessHoursId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Nullable, Update</p>
	<p>Description ID of the BusinessHours associated with the entitlement template. Must be a valid business hours ID.</p>
CasesPerEntitlement	<p>Type int</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description The total number of cases the entitlement template supports. This field is only available if <code>IsPerIncident</code> is <code>true</code>.</p>
IsPerIncident	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the entitlement template is limited to supporting a specific number of cases (<code>true</code>) or not (<code>false</code>).</p>
Name	<p>Type string</p> <p>Properties Create, Filter, idLookup, Update</p> <p>Description Required. Name of the entitlement template.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix_componentName</code> notation.</p>
	<p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the

Field	Details
	<p>namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.</p>
	<ul style="list-style-type: none"> In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
	<p>Available in version 34.0 and later.</p>
SlaProcessId	<p>Type reference</p>
	<p>Properties Create, Filter, Nullable, Update</p>
	<p>Description</p>
	<p>ID of the SlaProcess associated with the entitlement template. This field is available in API version 19.0 and later.</p>
Term	<p>Type int</p>
	<p>Properties Create, Defaulted on create, Filter, Nullable, Update</p>
	<p>Description</p>
	<p>Number of days that the entitlement template is valid.</p>
Type	<p>Type picklist</p>
	<p>Properties Create, Filter, Nullable, Update</p>
	<p>Description</p>
	<p>The type of entitlement template, such as Web or phone support.</p>

Usage

Use this object to manage entitlement templates.

EntityHistory

Represents historical information about an object's changed field values. This object is only available to users with the "View All Data" permission. This object is unavailable beginning with API version 8.0. Use the object-specific History objects instead: CaseHistory, ContractHistory, LeadHistory, OpportunityFieldHistory, OpportunityHistory ProcessInstanceHistory, QuantityForecastHistory, RevenueForecastHistory, or SolutionHistory.

Supported Calls

`describeSObjects(), getUpdated(), getDeleted(), query(), retrieve()`

Fields

Field	Details
FieldName	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description ID of the standard or custom field.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description New value of the modified field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description Previous value of the modified field.</p>
ParentId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the object that contains the field.</p>

Field	Details
ParentObjectType	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The kind of object that contains the field.</p>

Usage

In API version 7.0 and later, this object works with Case, Contract, and Solution objects:

- This object is always read-only in the online application.
- When a field is modified, this object records both the old and new field values. There are exceptions to this behavior for certain fields such as long text areas and multi-select picklists. These fields appear in this object to indicate that the field was changed, but the old and new values are not recorded.
- Two rows are added to this object when foreign key fields change. One row contains the foreign key object names that display in the online application. For example, "Jane Doe" is recorded as the name of a contact. The other row contains the actual foreign key ID that is only returned to and visible from the API.
- Up to a total of twenty fields (standard or custom) can be tracked for a given object.
- In the online application, you can specify which fields are tracked or not tracked at any time.
- As soon as tracking is turned on for a field, all changes to its value are recorded in the database.
- Turning off tracking for a field stops further changes from being recorded, but the history data is not deleted.
- Be advised that deleting a custom field also permanently deletes the history data for that custom field.

EntityMilestone

Represents a required step in a customer support process on a record. The Salesforce user interface uses the term "object milestone." This object is available in API version 37.0 and later.

As of Summer '16, object milestones are supported only for work orders.



Note: Milestones on cases use the [CaseMilestone](#) object type.

Supported Calls

```
delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update()
```

Special Access Rules

Object milestones are accessible only when entitlement management is enabled. You can add milestones to work orders only when work orders are enabled.

Fields

Field Name	Details
ActualElapsedTimeInDays	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The number of days that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)</p> <p> Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.</p>
ActualElapsedTimeInHrs	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The number of hours that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)</p> <p> Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.</p>
ActualElapsedTimeInMins	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of minutes that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)</p> <p> Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.</p>
BusinessHoursId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	<p>The business hours on the milestone. If business hours aren't specified, the entitlement process business hours are used. If business hours are also not specified on the entitlement process, the business hours on the record are used.</p>
CompletionDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort, Update
	Description
	The date and time the milestone was completed.
CurrencyIsoCode	Type
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.
Elapsed Time In Days	Type
	double
	Properties
	Filter, Nillable, Sort
	Description
	The number of days it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
Elapsed Time In Hours	Type
	double
	Properties
	Filter, Nillable, Sort
	Description
	The number of hours it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
Elapsed Time In Minutes	Type
	int

Field Name	Details
	Properties Filter, Group, Nillable, Sort
	Description The number of minutes it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
IsCompleted	Type boolean
	Properties Defaulted on create, Filter, Group, Sort
	Description Icon (✓) that indicates a milestone completion.
IsViolated	Type boolean
	Properties Defaulted on create, Filter, Group, Sort
	Description Icon (⚠) that indicates a milestone violation.
MilestoneTypeID	Type reference
	Properties Filter, Group, Nillable, Sort
	Description The ID of the milestone (for instance, First Response).
Name	Type string
	Properties Filter, Group, Sort, Update
	Description The name of the milestone.
ParentEntityID	Type reference
	Properties Filter, Group, Sort

Field Name	Details
	Description
	<p>The ID of the record—for example, a work order—that contains the milestone.</p>
StartDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort, Update
	Description
	<p>The date and time that milestone tracking started.</p>
StoppedTimeInDays	Type
	double
	Properties
	Filter, Nillable, Sort
	Description
	<p>The number of days that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.</p>
	 Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
StoppedTimeInHrs	Type
	double
	Properties
	Filter, Nillable, Sort
	Description
	<p>The number of hours that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.</p>
	 Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
StoppedTimeInMins	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The number of minutes that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.</p>

Field Name	Details
	<p> Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.</p>
TargetDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The date and time to complete the milestone.</p>
TargetResponseInDays	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The number of days to complete the milestone. Automatically calculated to include the business hours on the record.</p>
TargetResponseInHrs	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The number of hours to complete the milestone. Automatically calculated to include the business hours on the record.</p>
TargetResponseInMins	<p>Type int</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The number of minutes to complete the milestone. Automatically calculated to include the business hours on the record.</p>
TimeRemainingInDays	<p>Type string</p> <p>Properties Group, Nullable, Sort</p> <p>Description The days that remain before a milestone violation. Automatically calculated to include the business hours on the record.</p>

Field Name	Details
TimeRemainingInHrs	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The hours that remain before a milestone violation. Automatically calculated to include the business hours on the record.</p>
TimeRemainingInMins	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The minutes that remain before a milestone violation. Automatically calculated to include the business hours on the record.</p>
TimeSinceTargetInDays	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The days that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.</p>
TimeSinceTargetInHrs	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The hours that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.</p>
TimeSinceTargetInMins	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The minutes that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.</p>

Usage

When you create an entitlement process, you select its type based on the type of record that you want the process to run on: Case or Work Order. Processes created before Summer '16 use the Case type. When a Work Order entitlement process runs on a work order, the resulting milestones on the work order are object milestones. Conversely, when a Case entitlement process runs on a case, the resulting milestones are case milestones, a separate standard object.

 **Tip:** If an entitlement has an entitlement process associated with it, don't use the entitlement for multiple types of support records. An entitlement process works only on records that match the process's type. For example, when a Case entitlement process is applied to an entitlement, the process runs only on cases associated with that entitlement. If a work order is also associated with the entitlement, the process doesn't run on the work order. To ensure that the milestones you set up work as expected, associate a customer's work orders and cases with different entitlements.

Customize page layouts, validation rules, and more for object milestones from the Object Milestones node in Setup under Entitlement Management.

EntitySubscription

Represents a subscription for a user following a record or another user. This object is available in API version 34.0 and later.

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the community that this file originated from. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the record or user which the user is following.</p>

Field	Details
SubscriberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the User who is following the record or user.</p>

Usage

Consider this when following records and users:

- Users can only follow records that they can see.
- Users can see which records other users are following, unless they don't have access to the records.
- Administrators and users with the "Modify All Users" permission can configure a user to follow records that the user has read access to.
- Administrators and users with the "Modify All Users" permission can configure users to stop following records.
- Following topics is available in API version 29.0 and later. For this reason, a topic ID is now a supported value for the `ParentId` field.
- If you deactivate a user, any EntitySubscription where the user is associated with the `ParentId` or `SubscriberId` field, meaning all subscriptions both to and from the user, are soft deleted. If the user is reactivated, the subscriptions are restored. However, if you deactivate multiple users at once and these users follow each other, their subscriptions are hard deleted. In this case, the user-to-user EntitySubscription is deleted twice (double deleted). Such subscriptions can't be restored upon user reactivation.

When using `query()` with EntitySubscription,

- A query must include a `LIMIT` clause and the limit can't exceed 1000.
- A query using a `WHERE` clause can only filter by fields on the EntitySubscription object.
- If user sharing is enabled and the querying user is not an administrator, a SOQL query must be constrained either by the `ParentId` or `SubscriberId`. Otherwise, the query behavior at run time is undefined, meaning the result set can be incomplete or inconsistent from invocation to invocation. For an unconstrained query, the sharing check limits imposed on a non-administrative user are likely to be exceeded before the query completes, because access checks are run against both parent and subject, for each row of the result set. We recommend using the Chatter REST API to query EntitySubscription data instead of running a SOQL query.
- Users without the "View All Data" permission
 - Need read access on the object associated with the `ParentId` field to see which users are following records for the object.
 - Can use an `ORDER BY` clause in a query only to order by fields on the EntitySubscription object. For example, if the subscription relates to an Account record, the query can `ORDER BY ParentId`, but it can't `ORDER BY Account.Name`.
 - Don't always get all matching subscriptions when running a query. For these users, a query evaluates visibility criteria on a maximum of 500 records to reduce the prospect of long-running queries. If a user runs a query to see the CEO's subscriptions, it might scan a large number of records. The query only returns matches within the first 500 records scanned. It is possible that there are more subscriptions that are visible to the user, but they are not returned. To mitigate this, we recommend using a `WHERE` clause, if possible, to reduce the scope of the query.

Sample—SOQL

The following SOQL query returns subscriptions for all the accounts that a subscriber is following that have more than 10 employees:

```
SELECT Id  
FROM EntitySubscription  
WHERE SubscriberId = '005U0000000Rg2CIAS'  
AND ParentId IN (  
    SELECT Id FROM Account  
    WHERE NumberOfEmployees > 10  
)  
LIMIT 200
```

SEE ALSO:

[AccountFeed](#)
[CaseFeed](#)
[ContactFeed](#)
[OpportunityFeed](#)
[SolutionFeed](#)

EnvironmentHubMember

Represents a member organization in the Environment Hub. This object is available in API version 29.0 and later.



Note: You can only create 20 member orgs per day. If you need to create additional orgs, log a case in the Partner Community.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete()`

Fields

Field Name	Details
CreatedDate	Type dateTime Properties Defaulted on create, Filter, Sort Description System field that specifies the date and time when this record was created.
Description	Type string

Field Name	Details
	Properties
	Nillable, Update
	Description
	A brief description of this organization.
DisplayName	Type
	string
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	The name that the user has specified for this member organization.
EnvironmentHubId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	The Organization ID of this member's Environment Hub organization.
Id	Type
	ID
	Properties
	Defaulted on create, Filter, Group, idLookup, Sort
	Description
	System field that uniquely identifies this record.
IsFedIdSsoMatchAllowed	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates if single sign-on has been enabled based on matching Federation ID. The default value is <code>false</code> .
IsSandbox	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates if the member organization is a sandbox (<code>true</code>) or not (<code>false</code>). This field is available in API version 36.0 and later.

Field Name	Details
MemberEntity	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The unique Organization ID of the member organization for this record.</p>
MemberType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of member organization for this record. Possible values include Branch Org, Patch Org, Release Org, Sandbox Org, Trialforce Management Org, and Trialforce Source Org.</p> <p> Note: Only one member type at a time is stored. Member type is determined according to this hierarchy: (1) Sandbox, (2) Release, (3) Trialforce Source Org (TSO), (4) Patch, (5) Branch, and (6) Trialforce Management Org (TMO). For example, if an org is both a sandbox and a TMO, the value of MemberType is Sandbox Org.</p>
Name	<p>Type string</p> <p>Properties Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the member organization for this record.</p>
OrgEdition	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The organization's edition, for example, Enterprise Edition or Unlimited Edition.</p>
OrgStatus	<p>Type picklist</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The licensing or creation status of this organization. Possible values include Active, Demo, Deleted, Free, Inactive, and Trial.</p>

Field Name	Details
Origin	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The method by which this organization was added to the Environment Hub. Possible values are <code>Auto</code> <code>Discovered</code>, <code>User Added</code>, and <code>Provisioned</code>.</p>
ServiceProviderId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the service provider for this member organization. This field is available in API version 36.0 and later.</p>
ShouldAddRelatedOrgs	<p>Type boolean</p> <p>Properties Defaulted on create, Update</p> <p>Description If related organizations should be automatically connected to the hub when this member organization is added. The default value is <code>true</code>.</p>
ShouldEnableSSO	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If SSO should be enabled when this member organization is added. The default value is <code>false</code>.</p>
SSOMappedUsers	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description The total number of mapped users in this member organization. This field is available in API version 36.0 and later.</p>
SsoStatus	<p>Type picklist</p>

Field Name	Details
	<p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If single sign-on has been enabled for this organization. Possible values are:</p> <ul style="list-style-type: none"> • Enabled—Single sign-on is enabled. • Disabled—Single sign-on is disabled. • Pending—Single sign-on is in the process of being enabled. • Failed—Single sign-on enablement failed. Contact Salesforce support for assistance.
SsoUsernameFormula	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description The custom formula for matching users in the hub and member organizations.</p>

Usage

Use this object to access and modify settings of member organizations in the Environment Hub.

Event

Represents an event in the calendar. In the user interface, event and task records are collectively referred to as activities.

 **Note:**

- An EventRelation object can't be related to a child event, and child events don't include the invitee related list.
- `query()`, `delete()`, and `update()` aren't allowed with events related to more than one contact in API versions 25.0 and earlier.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AcceptedEventInviteeIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of contact or lead IDs who accepted this event. This JunctionIdList is linked to the AcceptedEventRelation child relationship.</p> <p> Warning: Adding a JunctionIdList field name to the fieldsToNull property deletes all related junction records. This action can't be undone.</p>
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the ID of the related Account. The AccountId is determined as follows. If the value of WhatId is any of the following objects, then Salesforce uses that object's AccountId.<ul style="list-style-type: none">• Account• Opportunity• Contract• Custom object that is a child of AccountIf the value of the WhatId field is any other object, and the value of the WhoId field is a Contact object, then Salesforce uses that contact's AccountId. (If your organization uses Shared Activities, Salesforce uses the AccountId of the primary contact.) Otherwise, Salesforce sets the value of the AccountId field to null. For information on IDs, see ID Field Type.</p>
ActivityDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contains the event's due date if the IsAllDayEvent flag is set to true. This field is a date field with a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. Don't attempt to alter the timestamp to account for time zone differences. Label is Due Date Only. This field is required in versions 12.0 and earlier if the IsAllDayEvent flag is set to true.</p>

Field	Details
	The value for this field and <code>StartTime</code> must match, or one of them must be <code>null</code> .
ActivityDateTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Contains the event's due date if the <code>IsAllDayEvent</code> flag is set to <code>false</code>. This field is a regular Date/Time field with a relevant time portion. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate. Label is Due Date Time. This field is required in versions 12.0 and earlier if the <code>IsAllDayEvent</code> flag is set to <code>false</code>. The value for this field and <code>StartTime</code> must match, or one of them must be <code>null</code>.</p>
ClientGuid	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The client globally unique identifier identifies the external API client used to create the event. Label is Client GUID.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
DeclinedEventInviteeIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of contact, lead, or user IDs who declined this event. This <code>JunctionIdList</code> is linked to the <code>DeclinedEventRelation</code> child relationship.</p> <p> Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.</p>

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Contains a text description of the event. Limit: 32,000 characters.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.</p>
DurationInMinutes	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contains the event length, in minutes. Note that even though this field represents a temporal value, it is an integer type—not a Date/Time type. This field is required in versions 12.0 and earlier if <code>IsAllDayEvent</code> is false. In versions 13.0 and later, this field is optional, depending on the following:</p> <ul style="list-style-type: none"> • If <code>IsAllDayEvent</code> is true, you can supply a value for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are <code>null</code>, the duration defaults to one day. • If <code>IsAllDayEvent</code> is false, a value must be supplied for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. <p>If the multiday event feature is enabled, then API versions 13.0 and later support values greater than 1440 for the <code>DurationInMinutes</code> field. API versions 12.0 and earlier can't access event objects whose <code>DurationInMinutes</code> is greater than 1440. For more information, see Multiday Events.</p>
EndDateTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
Description	Available in versions 13.0 and later. This field is a regular Date/Time field with a relevant time portion. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate.
Description	This field is optional, depending on the following:
<ul style="list-style-type: none"> • If <code>IsAllDayEvent</code> is true, you can supply a value for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are <code>null</code>, the duration defaults to one day. • If <code>IsAllDayEvent</code> is false, a value must be supplied for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. 	
EventSubtype	Type picklist
Properties	Create, Filter, Group, Nillable, Restricted picklist, Sort
Description	Provides standard subtypes to facilitate creating and searching for events. This field isn't updateable.
EventWhoIds	Type <code>JunctionIdList</code>
Properties	Create, Update
Description	A string array of contact or lead IDs used to create many-to-many relationships with a shared event. <code>EventWhoIds</code> is available when the shared activities setting is enabled. The first contact or lead ID in the list becomes the primary <code>WhoId</code> if you don't specify a primary <code>WhoId</code> .
 Warning: Adding a <code>JunctionIdList</code> field name to the <code>fieldsToNull</code> property deletes all related junction records. This action can't be undone.	
GroupEventType	Type picklist
Properties	Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort
Description	Read-only. Available in API versions 19.0 and later.
The possible values are:	<ul style="list-style-type: none"> • 0 (Non-group event)—An event with no invitees.

Field	Details
	<ul style="list-style-type: none">• 1 (Group event)—An event with invitees.• 2 (Proposed event)—An event created when a user requests a meeting with a contact, lead, or person account using the Salesforce user interface. When the user confirms the meeting, the proposed event becomes a group event. You can't create, edit, or delete proposed events in the API. <p>For more information, see How Cloud Scheduler Works in Salesforce Classic in the Salesforce help.</p>
IsAllDayEvent	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the <code>ActivityDate</code> field (<code>true</code>) or the <code>ActivityDateTime</code> field (<code>false</code>) is used to define the date or time of the event. Label is All-Day Event. See also DurationInMinutes and EndDateTime.</p>
IsArchived	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the event has been archived.</p>
IsChild	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the event is a child of another event (<code>true</code>) or not (<code>false</code>). For a child event, you can update <code>IsReminderSet</code> and <code>ReminderDateTime</code> only. You can query and delete a child event. If the objects related to the child event are different from those related to the parent event (this difference is possible if you use API version 25.0 or earlier) and one of the objects related to the child event is deleted, the objects related to the parent event are updated to ensure data integrity.</p>
IsClientManaged	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
Description	Indicates whether the event is managed by an external client. If the value of this field is false, the event isn't owned or managed by an external client, and Salesforce can be used to update it. If the value is true, Salesforce can be used to change only noncritical fields on the event. Label is Is Client Managed .
<code>IsGroupEvent</code>	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	Indicates whether the event is a group event—that is, whether it has invitees (<code>true</code>) or not (<code>false</code>).
<code>IsPrivate</code>	Type boolean
Properties	Create, Defaulted on create, Filter, Group, Sort, Update
Description	Indicates whether users other than the creator of the event can (<code>false</code>) or can't (<code>true</code>) see the event details when viewing the event user's calendar. However, users with the View All Data or Modify All Data permission can see private events in reports and searches, or when viewing other users' calendars. Private events can't be associated with opportunities, accounts, cases, campaigns, contracts, leads, or contacts. Label is Private .
<code>IsRecurrence</code>	Type boolean
Properties	Create, Defaulted on create, Filter, Group, Sort
Description	Indicates whether the event is scheduled to repeat itself (<code>true</code>) or only occurs once (<code>false</code>). This is a read-only field when updating records, but not when creating them. If this field value is <code>true</code> , then <code>RecurrenceEndDateOnly</code> , <code>RecurrenceStartTime</code> , <code>RecurrenceType</code> , and any recurrence fields associated with the given recurrence type must be populated. Label is Create recurring series of events .
<code>IsReminderSet</code>	Type boolean
Properties	Create, Defaulted on create, Filter, Group, Sort, Update
Description	Indicates whether the activity is a reminder (<code>true</code>) or not (<code>false</code>).

Field	Details
IsVisibleInSelfService	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether an event associated with an object can be viewed in the Customer Portal (<code>true</code>) or not (<code>false</code>). If your organization has enabled Communities, events marked <code>IsVisibleInSelfService</code> are visible to any external user in the community, as long as the user has access to the record the event was created on. This field is available when Customer Portal or partner portal are enabled OR Communities is enabled and you have Customer Portal or partner portal licenses.</p>
Location	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contains the location of the event.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Contains the ID of the user who owns the event. Label is Assigned to ID.</p>
RecurrenceActivityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read-only. Not required on create. Contains the ID of the main record of the recurring event. Subsequent occurrences have the same value in this field.</p>
RecurrenceDayOfMonth	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates the day of the month on which the event repeats.</p>

Field	Details
RecurrenceDayOfWeekMask	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates the day or days of the week on which the event repeats. This field contains a bitmask. The values are as follows:</p> <ul style="list-style-type: none"> • Sunday = 1 • Monday = 2 • Tuesday = 4 • Wednesday = 8 • Thursday = 16 • Friday = 32 • Saturday = 64 <p>Multiple days are represented as the sum of their numerical values. For example, Tuesday and Thursday = $4 + 16 = 20$.</p>
RecurrenceEndDateOnly	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates the last date on which the event repeats. For multiday recurring events, this is the day on which the last occurrence starts. This field is a date field with a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. Don't attempt to alter the timestamp to account for time zone differences.</p>
RecurrenceInstance	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the frequency of the event's recurrence. For example, 2nd or 3rd.</p>
RecurrenceInterval	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates the interval between recurring events.</p>

Field	Details
RecurrenceMonthOfYear	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the month in which the event repeats.</p>
RecurrenceStartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Indicates the date and time when the recurring event begins. The value must precede the <code>RecurrenceEndDateOnly</code>. This field is a regular Date/Time field with a relevant time portion. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate.</p>
RecurrenceTimeZoneSidKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the time zone associated with a recurring event. For example, "UTC-8:00" for Pacific Standard Time.</p>
RecurrenceType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates how often the event repeats. For example, daily, weekly, or every nth month (where "nth" is defined in <code>RecurrenceInstance</code>).</p>
ReminderDateTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Represents the time when the reminder is scheduled to fire, if <code>IsReminderSet</code> is set to <code>true</code>. If <code>IsReminderSet</code> is set to <code>false</code>, then the user may have deselected the</p>

Field	Details
	reminder checkbox in the Salesforce user interface, or the reminder has already fired at the time indicated by the value.
ShowAs	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates how this event appears when another user views the calendar: Busy, Out of Office, or Free. Label is Show Time As.</p>
StartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Indicates the start date and time of the event. Available in versions 13.0 and later.</p> <p>If the Event <code>IsAllDayEvent</code> flag is set to true (indicating that it is an all-day Event), then the event start date information is contained in the <code>StartTime</code> field. This field is a regular Date/Time field with a relevant time portion. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate.</p> <p>If the Event <code>IsAllDayEvent</code> flag is set to false (indicating that it is not an all-day event), then the event start date information is contained in the <code>StartTime</code> field. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. You need to translate the time portion to or from a local time zone for the user or the application, as appropriate.</p> <p>If this field has a value, then <code>ActivityDate</code> and <code>ActivityDateTime</code> must either be <code>null</code> or match the value of this field.</p>
Subject	<p>Type combobox</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The subject line of the event, such as Call, Email, or Meeting. Limit: 255 characters.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Update</p>

Field	Details
	<p>Description Indicates the event type, such as Call, Email, or Meeting.</p>
UndecidedEventInviteeIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of contact, lead, or user IDs who are undecided about this event. This JunctionIdList is linked to the UndecidedEventRelation child relationship.</p> <p> Warning: Adding a JunctionIdList field name to the fieldsToNull property deletes all related junction records. This action can't be undone.</p>
WhatCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Available if your organization has enabled Shared Activities. Represents the count of related EventRelations pertaining to the WhatId. The count of the WhatId must be 1 or less.</p>
WhatId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The WhatId represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. WhatIds are polymorphic. Polymorphic means a WhatId is equivalent to the ID of a related object. The label is Related To ID.</p>
WhoCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Available to organizations that have Shared Activities enabled. Represents the count of related EventRelations pertaining to the WhoId.</p>
WhoId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
Description	The Whold represents a human such as a lead or a contact. Wholds are polymorphic. Polymorphic means a Whold is equivalent to a contact's ID or a lead's ID. The label is Name ID.
	If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the Whold field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is Name ID.
	If the JunctionIdList field is used, all WhoIds are included in the relationship list.

Usage

Use Event to manage calendar appointments.

Querying and Filtering Events

Queries on events will be denied before they time out if they involve amounts of data that are deemed too large. In such cases, the exception code `OPERATION_TOO_LARGE` is returned. If you receive `OPERATION_TOO_LARGE`, refactor your query to return or scan a smaller amount of data.

When querying for events with a specific due date, you must filter on both the `ActivityDateTime` and `ActivityDate` fields. For example to find all events with a due date of February 14, 2003, you need two filters:

- One filter with the `ActivityDate` field equal to the Coordinated Universal Time (UTC) time zone on February 14, 2003.
- One filter with the `ActivityDateTime` field greater than or equal to midnight on February 14, 2003 in the user's local time zone AND less than or equal to midnight on February 15, 2003 in the user's local time zone.

Alternatively, in version 13.0 and later, you can find events with a specific due date by filtering on `StartTime`. For example, to find all events with a due date of February 14, 2003, filter with the `StartTime` greater than or equal to midnight on February 14, 2003 in the user's local time zone AND less than or equal to midnight on February 15, 2003 in the user's local time zone.

The `EventId` field of an `EventRelation` object always points to the master record. An invitee on a group event can query the `EventRelation` object to view the master record.

Multiday Events

- Multiday events are available in version 13.0 and later. Also, in earlier versions SOQL queries do not return multiday events.
- Multiday events are enabled through the user interface from Setup by entering `Activity Settings` in the Quick Find box, then selecting **Activity Settings**.
- If the multiday event feature is enabled, then API versions 13.0 and later support values greater than 1440 for the `DurationInMinutes` field. API versions 12.0 and earlier can't access event objects whose `DurationInMinutes` is greater than 1440.
- Multiday events can't exceed 14 days.

Recurring Events

- Recurring events are available in version 7.0 and later.
- After an event is created, it can't be changed from recurring to nonrecurring or vice versa.
- When you delete a recurring event series through the API, all past and future events in the series are removed. However, when you delete a recurring event series through the user interface, only future occurrences are removed.

- When creating a recurring event series, the duration of the event must be 24 hours or less (either the `DurationInMinutes` or the difference between `RecurrenceStartTime` and `EndDateTime` must be greater than 24 hours). Once the recurring event series is created, you can extend the length of individual occurrences beyond 24 hours if Multiday events are enabled; see **Multiday Events**.
- If `IsRecurrence` is `true`, then `RecurrenceStartTime`, `RecurrenceEndDateTime`, `RecurrenceType`, and any properties associated with the given recurrence type (see the following table) must be populated.
- When updating a recurring event series, it's not possible to update the `EventRelation` for the event series object and the `EventRelation` for the series object occurrences at the same time.

The following table describes the usage of recurrence fields. Each recurrence type must have all of its properties set. All unused properties must be set to null.

RecurrenceType Value	Properties	Example Pattern
RecurseDaily	RecurrenceInterval	Every second day
RecurseEveryWeekday	RecurrenceDayOfWeekMask	Every weekday - can't be Saturday or Sunday
RecurseMonthly	RecurrenceDayOfMonth RecurrenceInterval	Every second month, on the third day of the month
RecurseMonthlyNth	RecurrenceInterval RecurrenceInstance RecurrenceDayOfWeekMask	Every second month, on the last Friday of the month
RecurseWeekly	RecurrenceInterval RecurrenceDayOfWeekMask	Every three weeks on Wednesday and Friday
RecurseYearly	RecurrenceDayOfMonth RecurrenceMonthOfYear	Every March on the twenty-sixth day of the month
RecurseYearlyNth	RecurrenceDayOfWeekMask RecurrenceInstanceRecurrenceMonthOfYear	The first Saturday in every October

JunctionIdList

To create an event using `JunctionIdList`, IDs are pulled from the related contacts and both the event and the `EventRelation` records are created in one API call. If the `EventRelation` fails, the event is rolled back because it's all done in a single API call.

```
public void createEventNew(Contact[] contacts) {
    String[] contactIds = new String[contacts.size()];
    for (int i = 0; i < contacts.size(); i++) {
        contactIds[i] = contacts[i].getID();
    }
    Event event = new Event();
    event.setSubject("New Event");
    event.setEventWhoIds(contactIds);
    SaveResult[] results = null;
    try {
        results = connection.create(new Event[] {
            task
        });
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

```
}
```

SEE ALSO:

[Archived Activities](#)[Object Basics](#)

EventFeed

Represents a single feed item in the feed on an Event. This object is available in API version 20.0 and later.

Supported Calls

```
delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()
```

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of EventFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>

Field	Details
	<p>Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
	Description
	<p>Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	Type int
	Properties
	Group , Nullable , Sort
	Description
	<p>Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	Type string
	Properties
	Group , Nullable , Sort
	Description
	<p>Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	Type dateTime
	Properties
	Defaulted on create, Filter, Sort
	Description
	Date and time when this record was created. This field is a standard system field.
	Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.
FeedPostId	Type reference
	Properties
	Filter , Group , Nullable , Sort
	Description
	<p>This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only.</p>
	ID of the associated FeedPost . A FeedPost represents the following types of changes in an FeedItem : changes to tracked fields, text posts, link posts, and content posts.
InsertedById	Type reference

Field	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default <code>community</code>, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.

Field	Details
	<ul style="list-style-type: none">• AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope.• For feed items with a record parent, users can set NetworkScope only to AllNetworks.• You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the event record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none">• ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).

Field	Details
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none">• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RypplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• CaseCommentPost—generated event when a user adds a case comment for a case object• EmailMessageEvent—generated event when an email related to a case object is sent or received• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• ChangeStatusPost—generated event when a user changes the status of a case• AttachArticleEvent—generated event when a user attaches an article to a case <p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p>

Field	Details
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none"> • AllUsers—The feed item is available to all users who have permission to see the feed item. • InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none"> • For record posts, Visibility is set to InternalUsers for all internal users by default. • External users can set Visibility only to AllUsers. • On user and group posts, only internal users can set Visibility to InternalUsers.

Usage

Use this object to track changes for an event record. You can only delete a feed if you created it, or if you have the “Modify All Data” permission.

EventLogFile

Represents event log files for event monitoring. This object is available in API versions 32.0 and later.

The event monitoring feature allows you to gather information about your Salesforce organization’s operational events, which you can use to analyze usage trends and user behavior. You can interact with event monitoring data by querying fields on the EventLogFile object (like EventType and LogDate). To view the underlying event data, query theLogFile field. The EventType determines the schema of this field. See [Supported Event Types](#) for more information.

 **Note:** Log data schema for each EventType can change. At the start of each release, use the LogFileFieldNames and LogFileFieldTypes fields to validate schema changes for each EventType. In the unlikely case where no log files are generated for 24 hours, contact Salesforce.

Refer to the [Force.com REST API Developer’s Guide](#) for details about event monitoring.

Special Access Rules

Accessing this object requires View Event Log Files and API Enabled user permissions. Users with View All Data permission can view event log files.

Supported Calls

`query()`, `retrieve()`

Fields

Field	Details
EventType	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The event type—API, Login, Report, URI, and so forth. Use to determine which files were generated for your organization. See Supported Event Types for the corresponding <code>LogFile</code> schema.</p>
LogDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date and time of the log file's creation.</p>
LogFile	<p>Type base64</p> <p>Description Encoded file data in <code>.csv</code> format. The schema for this data is defined by the <code>EventType</code> field.</p>
LogFileContentType	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The content type of the log file; always <code>.csv</code>.</p>
LogFileFieldNames	<p>Type string</p> <p>Properties Nillable</p> <p>Description The ordered list of fields in the log file data.</p>

Field	Details
LogFileFieldTypes	<p>Type string</p> <p>Properties Nillable</p> <p>Description The ordered list of field types in the log file data (String, Id, and so forth).</p> <p> Note: LogFileFieldNames and LogFileFieldTypes are specific to each EventType. For example, LogFileFieldNames has a different value for an API EventType and a Login EventType.</p>
LogFileLength	<p>Type double</p> <p>Properties Filter, Sort</p> <p>Description The log file length in bytes. You can use this field to plan storage needs for your log files.</p>

Supported Event Types

The EventType field supports the following events.

- [Apex Callout](#)
- [Apex Execution](#)
- [Apex SOAP](#)
- [Apex Trigger](#)
- [API](#)
- [Asynchronous Report Run](#)
- [Bulk API](#)
- [Change Set Operation](#)
- [Console](#)
- [Content Distribution](#)
- [Content Document Link](#)
- [Content Transfer](#)
- [Dashboard](#)
- [Document Attachment Downloads](#)
- [Login As](#)

- [Login](#)
- [Logout](#)
- [Metadata API Operation](#)
- [Multiblock Report](#)
- [Package Install](#)
- [Queued Execution](#)
- [Report](#)
- [Report Export](#)
- [REST API](#)
- [Sandbox](#)
- [Sites](#)
- [Time-Based Workflow](#)
- [Transaction Security](#)
- [UI Tracking](#)
- [URI](#)
- [Visualforce Request](#)
- [Wave Change](#)
- [Wave Interaction](#)
- [Wave Performance](#)

The following tables serve as a reference for each event and its associated fields. Some common fields, such as CPU_TIME and RUN_TIME, can have null or zero values depending on how the events are generated for a given feature.

 **Note:** The Login and Logout events are available in supported Salesforce editions at no additional cost. Contact Salesforce to purchase the remaining event types.

Apex Callout

Apex Callout events contain details about callouts (external requests) during Apex code execution.

Field	Details
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the</p>

app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

METHOD**Type**

String

Description

The HTTP method of the callout.

Example

GET, POST, PUT, and so on.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDLk0FKfF5DV

REQUEST_SIZE**Type**

Number

	Description The size of the callout request body, in bytes.
RESPONSE_SIZE	Type Number Description The size of the callout response, in bytes.
RUN_TIME	Type Number Description Not used for this event type. Use the TIME field instead.
SESSION_KEY	Type Id Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page. Example d7DEq/ANa7nNZZVD
SUCCESS	Type Boolean Description 1 if the request was successful, and 0 if not.
TIME	Type Number Description The amount of time that the request took in milliseconds (ms).
TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

TYPE

Type

String

Description

The type of Apex callout.

Example

REST or AJAX

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

URL

Type

String

Description

The callout endpoint URL.

Example

www.salesforce.com

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

0053000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Apex Execution

Apex Execution events contain details about Apex classes that are used.

Field	Details
CALLOUT_TIME	<p>Type Number</p> <p>Description The execution time of the external Apex calls.</p>
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.</p>
DB_TOTAL_TIME	<p>Type Number</p> <p>Description The time in nanoseconds for a database round trip. Compare this field to <code>CPU_TIME</code> to determine whether performance issues are occurring in the database layer or in your own code.</p>
ENTRY_POINT	<p>Type String</p> <p>Description The entry point for this Apex execution.</p>

Example

- GeneralCloner.cloneAndInsertRecords
- VF- /apex/CloneUser

EVENT_TYPE

Type

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

EXEC_TIME

Type

Number

Description

The end-to-end Apex execution time.

LOGIN_KEY

Type

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

NUMBER_SOQL_QUESTIONS

Type

Number

Description

The number of SOQL queries that were executed during the event.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

QUIDDITY

Type

String

Description

The type of outer execution associated with this event.

Example

- A–Old Batch
- C–Scheduled Apex
- E–Inbound Email Service
- F–Future
- H–Apex REST
- I–Invocable Action
- K–Quick Action
- L–Aura
- M–Remote Action
- P–Parallel Batch Apex
- Q–Queuable
- R–Synchronous
- S–Serial Batch Apex
- T–Apex Tests
- V–Visualforce
- W–SOAP Webservices
- X–Execute Anonymous

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

RUN_TIME**Type**

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY**Type**

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Apex SOAP

Apex SOAP events contain details about Web Services API calls.

Field	Details
CLASS_NAME	<p>Type String</p> <p>Description The Apex class name. If the class is part of a managed package, this string includes the package namespace.</p>
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.</p>
DB_TOTAL_TIME	<p>Type Number</p> <p>Description The time in nanoseconds for a database round trip. Compare this field to <code>CPU_TIME</code> to determine whether performance issues are occurring in the database layer or in your own code.</p>
EVENT_TYPE	<p>Type String</p> <p>Description The type of event.</p> <p>Example ReportExport, URI, UITracking, API, RestApi, and so on.</p>
LIMIT_USAGE_PERCENT	<p>Type Number</p>

Description

The percentage of Apex SOAP calls that were made against the organization's limit.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

METHOD_NAME**Type**

String

Description

The name of the calling Apex method.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

QUERY**Type**

String

Description

The SOQL query, if one was performed.

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

REQUEST_STATUS**Type**

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
- F: Failure
- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

	Example /home/home.jsp
URI_ID_DERIVED	Type ID Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943
USER_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example 005900000001SNIA0

Apex Trigger

Apex Trigger events contain details about triggers that fire in an organization.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the

app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

DB_TOTAL_TIME**Type**

Number

Description

The time in nanoseconds for a database round trip. Compare this field to `CPU_TIME` to determine whether performance issues are occurring in the database layer or in your own code.

ENTITY_NAME**Type**

String

Description

The name of the object affected by the trigger.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

EXEC_TIME**Type**

Number

Description

The end-to-end Apex execution time.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example3nWgxWbDKWWDIk0FKfF5DV

REQUEST_STATUS**Type**

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
 - F: Failure
 - U: Undefined
 - A: Authorization Error
 - R: Redirect
 - N: Not Found
-

RUN_TIME**Type**

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY**Type**

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Exampled7DEq/ANa7nNZZVD

TIMESTAMP**Type**

String

Description

The access time of Salesforce services.

Example20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

TRIGGER_ID

Type

String

Description

The 15-character ID of the trigger that was fired.

TRIGGER_NAME

Type

String

Description

The name of the trigger that was fired.

TRIGGER_TYPE

Type

String

Description

The type of this trigger.

Possible Values

- AfterInsert
- AfterUpdate
- BeforeInsert
- BeforeUpdate

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

	Description The 15-character ID of the user who's using Salesforce services through the UI or the API.
	Example 00530000009M943
USER_ID_DERIVED	Type Id
	Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

API

API events contain details about your organization's Force.com Web Services API activity.

Field	Details
API_TYPE	Type String Description The type of API request. Possible Values <ul style="list-style-type: none"> • D: Apex Class • E: SOAP Enterprise • I: SOAP Cross Instance • L: Live Agent • M: SOAP Metadata • O: Old SOAP • P: SOAP Partner • R: REST API • S: SOAP Apex • T: SOAP Tooling • X: XmlRPC
API_VERSION	Type String Description The version of the API that's being used.

	Example 36.0
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services.
	Example 96.43.144.26
CLIENT_NAME	Type String Description The name of the client that's using Salesforce services.
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DB_BLOCKS	Type Number Description Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.
DB_CPU_TIME	Type Number Description Allows you to monitor trends in database uptime.
DB_TOTAL_TIME	Type Number Description The time in nanoseconds for a database round trip. Compare this field to CPU_TIME to determine whether performance issues are occurring in the database layer or in your own code.
ENTITY_NAME	Type Set

Description

API objects that are accessed.

Example

Account, Opportunity, Contact, and so on.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

METHOD_NAME**Type**

String

Description

The name of the calling Apex method.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWWDlk0FKff5DV

REQUEST_SIZE**Type**
Number**Description**

The size of the callout request body, in bytes.

REQUEST_STATUS**Type**
String**Description**

The status of the request for a page view or user interface action.

Possible Values

- S: Success
 - F: Failure
 - U: Undefined
 - A: Authorization Error
 - R: Redirect
 - N: Not Found
-

RESPONSE_SIZE**Type**
Number**Description**

The size of the callout response, in bytes.

ROWS_PROCESSED**Type**
Number**Description**

The number of rows that were processed in the request.

Example

150

RUN_TIME**Type**
Number**Description**

The amount of time that the request took in milliseconds.

QUERY**Type**
String**Description**

The SOQL query, if one was performed.

SESSION_KEY**Type**
Id

	Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.
	Example d7DEq/ANa7nNZZVD
TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ) Example 2015-07-27T11:32:59.555Z
URI	Type String Description The URI of the page that's receiving the request. Example /home/home.jsp
URI_ID_DERIVED	Type ID Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Asynchronous Report Run

Asynchronous Report Run events are created for reporting requests that are scheduled. This category includes dashboard refreshes, asynchronous reports, schedule reports, and analytics snapshots.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DASHBOARD_ID	Type String Description The 15-character ID of the dashboard that was run.
DASHBOARD_ID_DERIVED	Type String Description The 18-character case insensitive ID of the dashboard that was run.
DB_TOTAL_TIME	Type Number

	Description	
	The time in nanoseconds for a database round trip. Compare this field to <code>CPU_TIME</code> to determine whether performance issues are occurring in the database layer or in your own code.	
<code>DB_BLOCKS</code>	Type Number	
	Description	Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.
<code>DB_CPU_TIME</code>	Type Number	
	Description	Allows you to monitor trends in database uptime.
<code>DISPLAY_TYPE</code>	Type String	
	Description	The report display type, indicating the run mode of the report.
	Possible Values	<ul style="list-style-type: none">• <code>D</code>: Dashboard• <code>S</code>: Show Details• <code>H</code>: Hide Details
<code>ENTITY_NAME</code>	Type String	
	Description	The name of the object affected by the trigger.
<code>EVENT_TYPE</code>	Type String	
	Description	The type of event.
	Example	ReportExport, URI, UITracking, API, RestApi, and so on.
<code>LOGIN_KEY</code>	Type Id	

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

NUMBER_BUCKETS

Type

Number

Description

The number of buckets that were used in the report.

NUMBER_COLUMNS

Type

Number

Description

The number of columns in the report.

NUMBER_EXCEPTION_FILTERS

Type

Number

Description

The number of exception filters that are used in the report.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

RENDERING_TYPE

Type

String

Description

The report rendering type, describing the format of the report output.

Possible Values

- `w`: Web (HTML)
- `E`: Email
- `P`: Printable
- `x`: Excel
- `c`: Comma-separated values (CSV)
- `j`: JavaScript Object Notation (JSON)

REPORT_ID	Type Id Description The 15-character ID of the report that was run.
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV
REQUEST_STATUS	Type String Description The status of the request for a page view or user interface action. Possible Values <ul style="list-style-type: none">• S: Success• F: Failure• U: Undefined• A: Authorization Error• R: Redirect• N: Not Found
RUN_TIME	Type Number Description The amount of time that the request took in milliseconds.
SESSION_KEY	Type Id Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page. Example d7DEq/ANa7nNZZVD
SORT	Type String

Description

The sort column and order that was used in the report.

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

0053000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Bulk API

Bulk API events contain details about Bulk API requests.

Field	Details
BATCH_ID	<p>Type String</p> <p>Description The 15-character ID of the Bulk API batch.</p>
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.</p>
ENTITY_TYPE	<p>Type String</p> <p>Description The type of entity that the Bulk API used.</p>
EVENT_TYPE	<p>Type String</p> <p>Description The type of event.</p> <p>Example ReportExport, URI, UITracking, API, RestApi, and so on.</p>

JOB_ID	Type String Description The 15-character ID of the Bulk API job.
LOGIN_KEY	Type Id Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring. Example GeJCsym5eyvtEK2I
MESSAGE	Type EscapedString Description Any success or error message that's associated with the request.
NUMBER_FAILURES	Type Number Description The number of failures that were returned with the request.
OPERATION_TYPE	Type String Description The type of Bulk API operation that was performed.
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

ROWS_PROCESSED

Type

Number

Description

The number of rows that were processed in the request.

Example

150

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

SUCCESS

Type

Boolean

Description

Whether the batch was successful.

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

0053000009M943

Change Set Operation

Change Set Operation events contain information from change set migrations.

Field	Details
CHANGE_SET_NAME	Type String Description The name of the change set.
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number

Description

The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

OPERATION**Type**

String

Description

The operation that's being performed.

Possible Values

- DEPLOY
 - RETRIEVE
 - LIST
 - DESCRIBE
-

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D00000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TARGET_ORG_ID

Type

Id

Description

The 15-character ID of the organization that's receiving the change set.

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URL of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

00590000000I1SNIA0

Console

Console events contain information about the performance and use of Salesforce Consoles. The Console events are logged whenever a Console tab is opened with a sidebar component. Outside of that, when Console tabs are opened, a regular view record detail event is served just like in Salesforce Classic.

Field	Details
CLIENT_IP	<div data-bbox="816 1562 881 1596" data-label="Section-Header">Type</div> <div data-bbox="851 1598 887 1630" data-label="Text">IP</div> <div data-bbox="816 1647 953 1681" data-label="Section-Header">Description</div> <div data-bbox="851 1683 1429 1717" data-label="Text"> <p>The IP address of the client that's using Salesforce services.</p> </div> <div data-bbox="816 1727 923 1763" data-label="Section-Header">Example</div> <div data-bbox="851 1765 988 1797" data-label="Text"> <p>96.43.144.26</p> </div>

COMPONENT_ID	Type Id	Description The 15-character ID of the component.
COMPONENT_ID_DERIVED	Type Id	Description The 18-character, case-insensitive ID of the component.
CONSOLE_ID	Type Id	Description The 15-character ID of the console.
CONSOLE_ID_DERIVED	Type Id	Description The 18-character, case-insensitive ID of the console.
CPU_TIME	Type Number	Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DB_TOTAL_TIME	Type Number	Description The time in nanoseconds for a database round trip. Compare this field to CPU_TIME to determine whether performance issues are occurring in the database layer or in your own code.
EVENT_TYPE	Type String	Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.
LOGIN_KEY	Type Id	

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D00000000123

RECORD_ID

Type

Id

Description

The 15-character ID of the record that's associated with the console.

RECORD_ID_DERIVED

Type

Id

Description

The 18-character, case-insensitive ID of the record that's associated with the console.

RELATED_ENTITY_ID

Type

Id

Description

The 15-character ID of the record that's associated with the delivery distribution.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWWDlk0FKfF5DV

REQUEST_STATUS

Type

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
- F: Failure
- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

2013-07-15T23:33:22.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URL of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

00590000000I1SNIA0

Content Distribution

Content Distribution events contain information about content distributions and deliveries to users.

Field	Details
ACTION	Type String Description The action that's used when a delivery is viewed. Possible Values <ul style="list-style-type: none">• VIEW• INSERT• UPDATE

DELIVERY_ID**Type**

Id

Description

The 15-character ID of the content delivery.

DELIVERY_LOCATION**Type**

String

Description

The location of the delivery.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

RELATED_ENTITY_ID**Type**

Id

Description

The 15-character ID of the record that's associated with the delivery distribution.

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWWDlk0FKfF5DV

TIMESTAMP**Type**

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

VERSION_ID

Type

Id

Description

The 15-character ID of the content version.

Content Document Link

Content Document Link events contain sharing information for content documents.

Field	Details
DOCUMENT_ID	Type Id Description The 15-character ID of the document that's being shared.
EVENT_TYPE	Type String Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

SHARED_WITH_ENTITY_ID**Type**

Id

Description

Who the document was shared with.

SHARING_OPERATION**Type**

String

Description

The type of sharing operation on the document.

Possible Values

- INSERT
 - UPDATE
 - DELETE
-

SHARING_PERMISSION**Type**

String

Description

What permissions the document was shared with.

Possible Values

- v: Viewer
 - c: Collaborator
 - i: Inferred—that is, the sharing permissions were inferred from a relationship between the viewer and document. For example, a document's owner has a sharing permission to the document itself. Or, a document can be a part of a content collection, and the viewer has sharing permissions
-

to the collection rather than explicit permissions to the document directly.

TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ) Example 2015-07-27T11:32:59.555Z
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 0053000009M943

Content Transfer

Content Transfer events contain information about content transfer events, such as downloads, uploads, and previews.

Field	Details
DOCUMENT_ID	Type Id Description The 15-character ID of the document that's being shared.
DOCUMENT_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the document that's being shared.

EVENT_TYPE

Type
String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

FILE_PREVIEW_TYPE

Type
String

Description

The content type of the file version.

FILE_TYPE

Type
String

Description

The content type of the file preview.

ORGANIZATION_ID

Type
Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID

Type
String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

SIZE_BYTES

Type
Number

Description

The size of the file transfer, in bytes.

TIMESTAMP

Type
String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

TRANSACTION_TYPE

Type

String

Description

The operation that was performed.

Possible Values

- VersionDownloadAction and VersionDownloadApi represent downloads via the user interface and API respectively.
- VersionRenditionDownload represents a file preview action.
- saveVersion represents a file that's being uploaded.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

VERSION_ID

Type

Id

Description

The 15-character ID of the content version.

VERSION_ID_DERIVED	Type Id	Description The 18-character case insensitive ID of the content version.
--------------------	-------------------	--

Dashboard

Dashboard events contain details about dashboards that users view.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DASHBOARD_COMPONENT_ID	Type Id Description The 15-character ID of the dashboard component.
DASHBOARD_ID	Type String Description The 15-character ID of the dashboard that was run.
DASHBOARD_ID_DERIVED	Type String Description The 18-character case insensitive ID of the dashboard that was run.

DASHBOARD_TYPE**Type**

String

Description

The type of dashboard.

Possible Values

- R: Run as running user
 - C: Run as context user
 - S: Run as specific user
-

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

IS_SCHEDULED**Type**

Boolean

Description

The value is true if the dashboard is a scheduled dashboard.

IS_SUCCESS**Type**

Boolean

Description

1 if the dashboard component ran successfully, 0 if it didn't.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REPORT_ID	Type Id Description The 15-character ID of the report that was run.
REPORT_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the report that was run.
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV
RUN_TIME	Type Number Description The amount of time that the request took in milliseconds.
SESSION_KEY	Type Id Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page. Example d7DEq/ANa7nNZZVD
TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime

	Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)
	Example 2015-07-27T11:32:59.555Z
URI	Type String Description The URI of the page that's receiving the request. Example /home/home.jsp
URI_ID_DERIVED	Type ID Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943
USER_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example 005900000001SNIA0

Document Attachment Downloads

Document Attachment Downloads events contain details of document and attachment downloads.

Field	Details
CLIENT_IP	Type IP

Description

The IP address of the client that's using Salesforce services.

Example

96.43.144.26

ENTITY_ID**Type**

Id

Description

The 15-character ID of the entity that's associated with the document or attachment.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

FILE_NAME**Type**

String

Description

The name of the file or attachment.

FILE_TYPE**Type**

String

Description

The type of the file or attachment.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

Login As

Login As events contain details about what an administrator did while logged in as another user.

Field	Details
CLIENT_IP	<div data-bbox="816 1351 881 1383" data-label="Section-Header">Type</div> <div data-bbox="855 1385 886 1415" data-label="Text">IP</div> <div data-bbox="816 1431 954 1463" data-label="Section-Header">Description</div> <div data-bbox="855 1465 1432 1497" data-label="Text">The IP address of the client that's using Salesforce services.</div> <div data-bbox="816 1512 922 1543" data-label="Section-Header">Example</div> <div data-bbox="855 1545 990 1577" data-label="Text">96.43.144.26</div>
DELEGATED_USER_ID	<div data-bbox="816 1626 881 1657" data-label="Section-Header">Type</div> <div data-bbox="855 1660 886 1689" data-label="Text">Id</div> <div data-bbox="816 1706 954 1738" data-label="Section-Header">Description</div> <div data-bbox="855 1740 1457 1845" data-label="Text">The 15-character ID of the user who's using Salesforce services through the UI or API. In this case, the user who's doing the impersonation.</div>

DELEGATED_USER_ID_DERIVED

Type

Id

Description

The 18-character case-insensitive ID of the user who's using Salesforce services through the UI or API. In this case, the user who's doing the impersonation.

DELEGATED_USER_NAME

Type

String

Description

The username of the user who's using Salesforce services through the UI or API. In this case, the user who's doing the impersonation.

EVENT_TYPE

Type

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D00000000123

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Login

Login events contain details about your organization's user login history.

Field	Details
API_TYPE	<div data-bbox="816 1664 881 1695" data-label="Section-Header">Type</div> <div data-bbox="855 1698 923 1727" data-label="Text">String</div> <div data-bbox="816 1742 953 1776" data-label="Section-Header">Description</div> <div data-bbox="855 1778 1096 1809" data-label="Text"> <p>The type of API request.</p> </div>

Possible Values

- D: Apex Class
- E: SOAP Enterprise
- I: SOAP Cross Instance
- L: Live Agent
- M: SOAP Metadata
- O: Old SOAP
- P: SOAP Partner
- R: REST API
- S: SOAP Apex
- T: SOAP Tooling
- X: XMLRPC

API_VERSION**Type**

String

Description

The version of the API that's being used.

Example

36.0

BROWSER_TYPE**Type**

String

Description

The browser used for login.

Example Values

- 10011000: Internet Explorer Desktop 11
- 10011001: Internet Explorer Mobile 11
- 11035000: Firefox Desktop 35
- 11035001: Firefox Mobile 35
- 13050000: Chrome Desktop 50
- 13050001: Chrome Mobile 50
- 14012000: Safari Desktop 12
- 14012001: Safari Mobile 12

CIPHER_SUITE**Type**

String

Description

The TLS cipher suite used for the login. Values are OpenSSL-style cipher suite names, with hyphen delimiters. For more information, see [OpenSSL Cryptography and SSL/TLS Toolkit](#).

CPU_TIME**Type**
Number**Description**

The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

CLIENT_IP**Type**
IP**Description**

The IP address of the client that's using Salesforce services.

Example

96.43.144.26

DB_TOTAL_TIME**Type**
Number**Description**

The time in nanoseconds for a database round trip. Compare this field to `CPU_TIME` to determine whether performance issues are occurring in the database layer or in your own code.

EVENT_TYPE**Type**
String**Description**

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY**Type**
Id**Description**

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID**Type**
Id**Description**

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example3nWgxWbDKWWDIk0FKfF5DV

REQUEST_STATUS**Type**

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
 - F: Failure
 - U: Undefined
 - A: Authorization Error
 - R: Redirect
 - N: Not Found
-

RUN_TIME**Type**

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY**Type**

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Exampled7DEq/ANa7nNZZVD

SOURCE_IP**Type**

IP

Description

The source IP of the login request.

TIMESTAMP**Type**String

	<p>Description The access time of Salesforce services.</p> <p>Example 20130715233322.670</p>
TIMESTAMP_DERIVED	<p>Type Datetime</p> <p>Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)</p> <p>Example 2015-07-27T11:32:59.555Z</p>
TLS_PROTOCOL	<p>Type String</p> <p>Description The TLS protocol used for the login.</p> <p>Example There are 3 possible values.</p> <ul style="list-style-type: none">• 1.0• 1.1• 1.2
URI	<p>Type String</p> <p>Description The URI of the page that's receiving the request.</p> <p>Example /home/home.jsp</p>
URI_ID_DERIVED	<p>Type ID</p> <p>Description The 18-character case insensitive ID of the URI of the page that's receiving the request.</p>
USER_ID	<p>Type Id</p> <p>Description The 15-character ID of the user who's using Salesforce services through the UI or the API.</p> <p>Example 00530000009M943</p>

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

USER_NAME

Type

String

Description

The username that's used for login.

Logout

Logout events contain details of user logouts.

Field	Details
API_TYPE	<p>Type String</p> <p>Description The type of API request.</p> <p>Possible Values</p> <ul style="list-style-type: none"> • D: Apex Class • E: SOAP Enterprise • I: SOAP Cross Instance • L: Live Agent • M: SOAP Metadata • O: Old SOAP • P: SOAP Partner • R: REST API • S: SOAP Apex • T: SOAP Tooling • X: XmlRPC
API_VERSION	<p>Type String</p> <p>Description The version of the API that's being used.</p>

Example

36.0

APP_TYPE**Type**

Number

Description

The application type that was in use upon logging out.

Example Values

- 1007: SFDC Application
 - 1014: Live Agent
 - 2501: CTI
 - 2514: OAuth
 - 3475: SFDC Partner Portal
-

BROWSER_TYPE**Type**

String

Description

The browser used for login.

Example Values

- 10011000: Internet Explorer Desktop 11
 - 10011001: Internet Explorer Mobile 11
 - 11035000: Firefox Desktop 35
 - 11035001: Firefox Mobile 35
 - 13050000: Chrome Desktop 50
 - 13050001: Chrome Mobile 50
 - 14012000: Safari Desktop 12
 - 14012001: Safari Mobile 12
-

CLIENT_IP**Type**

IP

Description

The IP address of the client that's using Salesforce services.

Example

96.43.144.26

CLIENT_VERSION**Type**

Number

Description

The version of the client that was in use upon logging out.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

ORGANIZATION_ID**Type**

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

PLATFORM_TYPE**Type**

Number

Description

The code for the client platform. If a timeout caused the logout, this field is null.

Example Values

- 1000: Windows
- 2003: Macintosh/Apple OSX
- 5005: Android
- 5006: iPhone
- 5007: iPad

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDLk0FKfF5DV

RESOLUTION_TYPE**Type**

Number

Description

The screen resolution of the client. If a timeout caused the logout, this field is null.

SESSION_LEVEL**Type**

String

Description

The security level of the session that was used when logging out.

Possible Values

- 1: Standard Session
- 2: High-Assurance Session

SESSION_TYPE

Type

String

Description

The session type that was used when logging out.

Possible Values

- A: API
- I: APIOnlyUser
- N: ChatterNetworks
- Z: ChatterNetworksAPIOnly
- C: Content
- P: OauthApprovalUI
- O: Oauth2
- T: SiteStudio
- R: SitePreview
- S: SubstituteUser
- B: TempContentExchange
- G: TempOauthAccessTokenFrontdoor
- Y: TempVisualforceExchange
- F: TempUIFrontdoor
- U: UI
- E: UserSite
- V: Visualforce
- W: WDC_API

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322 . 670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_INITIATED_LOGOUT

Type

Boolean

Description

The value is 1 if the user intentionally logged out of the organization by clicking the **Logout** button. If the user's session timed out due to inactivity or another implicit logout action, the value is 0.

USER_NAME

Type

String

Description

The username that's used for login.

USER_TYPE

Type

String

Description

The category of user license of the user that logged out.

Possible Values

- S: Standard
- P: Partner
- p: Customer Portal Manager
- c: Customer Portal User
- o: Power Custom
- o: Custom
- L: Package License Manager
- x: Black Tab User
- N: Salesforce to Salesforce
- G: Guest

- D: External Who
- A: Automated Process
- b: High Volume Portal
- n: CSN Only
- F: Self Service

Metadata API Operation

Metadata API Operation events contain details of Metadata API retrieval and deployment requests.

Field	Details
API_VERSION	<p>Type String</p> <p>Description The version of the API that's being used.</p> <p>Example 36.0</p>
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.</p>
EVENT_TYPE	<p>Type String</p> <p>Description The type of event.</p> <p>Example ReportExport, URI, UITracking, API, RestApi, and so on.</p>
LOGIN_KEY	<p>Type Id</p>

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

OPERATION

Type

String

Description

The operation that's being performed.

Possible Values

- DEPLOY
- RETRIEVE
- LIST
- DESCRIBE

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Multiblock Report

Multiblock Report events contain details about Joined Report reports.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DB_TOTAL_TIME	Type Number Description The time in nanoseconds for a database round trip. Compare this field to CPU_TIME to determine whether performance issues are occurring in the database layer or in your own code.
EVENT_TYPE	Type String Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.

HAS_CHART	Type Boolean Description True if the report has a chart.
LOGIN_KEY	Type Id Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring. Example GeJCsym5eyvtEK2I
MASTER_REPORT_ID	Type String Description The 15-character ID of the master report.
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV
REQUEST_STATUS	Type String Description The status of the request for a page view or user interface action. Possible Values <ul style="list-style-type: none">• S: Success• F: Failure

- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

RUN_TIME	Type Number	Description The amount of time that the request took in milliseconds.
SESSION_KEY	Type Id	Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.
TIMESTAMP	Type String	Description The access time of Salesforce services.
	Example d7DEq/ANa7nNZZVD	
TIMESTAMP_DERIVED	Type Datetime	Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)
	Example 20130715233322.670	
URI	Type String	Description The URI of the page that's receiving the request.
	Example /home/home.jsp	
URI_ID_DERIVED	Type ID	

	Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 0053000000M943
USER_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example 0059000000I1SNIA0

Package Install

Package Install events contain details about package installation in the organization.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
EVENT_TYPE	Type String

	Description The type of event.
	Example ReportExport, URI, UITracking, API, RestApi, and so on.
FAILURE_TYPE	Type String Description A general categorization of any error that's encountered.
IS_MANAGED	Type Boolean Description True if the operation is performed on a managed package.
IS_PUSH	Type Boolean Description True if the package was installed as a result of a push upgrade.
IS_RELEASED	Type Boolean Description True if the operation is performed on a released package.
IS_SUCCESSFUL	Type Boolean Description True if the package was successfully installed.
LOGIN_KEY	Type Id Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring. Example GeJCsym5eyvtEK2I
OPERATION_TYPE	Type String Description The type of package operation.

Possible Values

- INSTALL
- UPGRADE
- EXPORT
- UNINSTALL
- VALIDATE_PACKAGE
- INIT_EXPORT_PKG_CONTROLLER

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

PACKAGE_NAME

Type

String

Description

The name of the package that's being installed.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDDlk0FKfF5DV

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

Queued Execution

Queued Execution events contain details about queued executions—for example, batch Apex.

Field	Details
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds that it took to complete the batch apex request. This field indicates the amount of activity taking place in the app server layer, allowing you to identify pieces of Apex or Visualforce code that need refactoring.</p>
DB_TOTAL_TIME	<p>Type Number</p> <p>Description The time in nanoseconds for a database round trip. Compare this field to <code>CPU_TIME</code> to determine whether performance issues are occurring in the database layer or in your own code.</p>
ENTRY_POINT	<p>Type String</p> <p>Description The name of the Apex class that serves as the execution point for the batch job.</p> <p>Example TaskPhoneExtensionBatchUpdate</p>
EVENT_TYPE	<p>Type String</p> <p>Description The type of event.</p> <p>Example ReportExport, URI, UITracking, API, RestApi, and so on.</p>
JOB_ID	<p>Type String</p>

Field	Details
	<p>Description The ID of the batch Apex job.</p> <p>Example 7073000000lDquo</p>
LOGIN_KEY	<p>Type Id</p> <p>Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.</p> <p>Example GeJCsym5eyvtEK2I</p>
ORGANIZATION_ID	<p>Type Id</p> <p>Description The 15-character ID of the organization.</p> <p>Example 00D000000000123</p>
REQUEST_ID	<p>Type String</p> <p>Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.</p> <p>Example 3nWgxWbDKWWDIk0FKfF5DV</p>
RUN_TIME	<p>Type Number</p> <p>Description The amount of time that the request took in milliseconds.</p>
SESSION_KEY	<p>Type Id</p> <p>Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.</p>

Field	Details
	Example d7DEq/ANa7nNZZVD
TIMESTAMP	Type String
	Description The access time of Salesforce services.
	Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime
	Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)
	Example 2015-07-27T11:32:59.555Z
URI	Type String
	Description The URI of the page that's receiving the request.
	Example /home/home.jsp
URI_ID_DERIVED	Type ID
	Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id
	Description The 15-character ID of the user who's using Salesforce services through the UI or the API.
	Example 00530000009M943

Report

Report events contain information about what happened when a user ran a report.

Field	Details
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
CPU_TIME	<p>Type Number</p> <p>Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.</p>
DB_BLOCKS	<p>Type Number</p> <p>Description Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.</p>
DB_CPU_TIME	<p>Type Number</p> <p>Description Allows you to monitor trends in database uptime.</p>
DB_TOTAL_TIME	<p>Type Number</p> <p>Description The time in nanoseconds for a database round trip. Compare this field to <code>CPU_TIME</code> to determine whether performance issues are occurring in the database layer or in your own code.</p>
DISPLAY_TYPE	<p>Type String</p> <p>Description The report display type, indicating the run mode of the report.</p>

Possible Values

- D: Dashboard
- S: Show Details
- H: Hide Details

ENTITY_NAME

Type

String

Description

The name of the object affected by the trigger.

EVENT_TYPE

Type

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY

Type

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

NUMBER_BUCKETS

Type

Number

Description

The number of buckets that were used in the report.

NUMBER_COLUMNS

Type

Number

Description

The number of columns in the report.

NUMBER_EXCEPTION_FILTERS

Type

Number

Description

The number of exception filters that are used in the report.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

RENDERING_TYPE

Type

String

Description

The report rendering type, describing the format of the report output.

Possible Values

- `w`: Web (HTML)
- `E`: Email
- `P`: Printable
- `X`: Excel
- `C`: Comma-separated values (CSV)
- `J`: JavaScript Object Notation (JSON)

REPORT_ID

Type

Id

Description

The 15-character ID of the report that was run.

REPORT_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the report that was run.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWWDlk0FKfF5DV

REQUEST_STATUS

Type

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
- F: Failure
- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

SORT

Type

String

Description

The sort column and order that was used in the report.

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

0053000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

00590000000I1SNIA0

Report Export

Report Export events contain details about reports that a user exported.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26

CLIENT_INFO	Type String Description Information about the client that's using Salesforce services.
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
EVENT_TYPE	Type String Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.
LOGIN_KEY	Type Id Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring. Example GeJCsym5eyvtEK2I
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
REPORT_DESCRIPTION	Type String Description Information about the report that was run.
REQUEST_ID	Type String

	Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.
	Example 3nWgxWbDKWWDIk0FKfF5DV
RUN_TIME	Type Number Description The amount of time that the request took in milliseconds.
SESSION_KEY	Type Id Description The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.
	Example d7DEq/ANa7nNZZVD
TIMESTAMP	Type String Description The access time of Salesforce services.
	Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)
	Example 2015-07-27T11:32:59.555Z
URI	Type String Description The URI of the page that's receiving the request.
	Example /home/home.jsp

URI_ID_DERIVED	Type ID	Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id	Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943
USER_ID_DERIVED	Type Id	Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example 005900000001SNIAO

REST API

REST API events contain details about REST-specific requests.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

DB_BLOCKS**Type**
Number**Description**

Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.

DB_CPU_TIME**Type**
Number**Description**

Allows you to monitor trends in database uptime.

DB_TOTAL_TIME**Type**
Number**Description**

The time in nanoseconds for a database round trip. Compare this field to `CPU_TIME` to determine whether performance issues are occurring in the database layer or in your own code.

ENTITY_NAME**Type**
Set**Description**

API objects that are accessed.

Example

Account, Opportunity, Contact, and so on.

EVENT_TYPE**Type**
String**Description**

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY**Type**
Id**Description**

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

MEDIA_TYPE**Type**
String

	Description The media type of the response.
METHOD	Type String Description The HTTP method of the request—GET, POST, PUT, and so on.
NUMBER_FIELDS	Type Number Description The number of fields or columns, where applicable.
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
REQUEST_STATUS	Type String Description The status of the request for a page view or user interface action. Possible Values <ul style="list-style-type: none">• S: Success• F: Failure• U: Undefined• A: Authorization Error• R: Redirect• N: Not Found
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV

ROWS_PROCESSED

Type
Number

Description

The number of rows that were processed in the request.

Example
150

RUN_TIME

Type
Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type
Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example
d7DEq/ANa7nNZVD

STATUS_CODE

Type
Number

Description

The HTTP status code for the response.

TIMESTAMP

Type
String

Description

The access time of Salesforce services.

Example
20130715233322.670

TIMESTAMP_DERIVED

Type
Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example
2015-07-27T11:32:59.555Z

URI

Type
String

	Description The URI of the page that's receiving the request.
	Example <code>/home/home.jsp</code>
URI_ID_DERIVED	Type ID Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_AGENT	Type Number Description The numeric code for the type of client used to make the request (for example, the browser, application, or API).
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example <code>0053000009M943</code>
USER_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example <code>00590000001SNIAO</code>

Sandbox

Sandbox events contain details about sandbox copies.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services.

	Example 96.43.144.26
CURRENT_SANDBOX_ORG_ID	Type Id Description The 15-character ID of the current sandbox organization.
EVENT_TYPE	Type String Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D00000000123
PENDING_SANDBOX_ORG_ID	Type Id Description The 15-character ID of the target sandbox org.
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV
SANDBOX_ID	Type Id Description The 15-character ID of the sandbox organization.
STATUS	Type String

	Description The status of the sandbox copy.
TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ) Example 2015-07-27T11:32:59.555Z
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943

Sites

Sites events contain details of Site.com requests. Requests can originate from the browser (UI) or the API.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number

Description

The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

DB_TOTAL_TIME

Type

Number

Description

The time in nanoseconds for a database round trip. Compare this field to CPU_TIME to determine whether performance issues are occurring in the database layer or in your own code.

EVENT_TYPE

Type

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

HTTP_HEADERS

Type

String

Description

The HTTP headers that were sent in the request.

METHOD

Type

String

Description

The HTTP method of the request—GET, POST, PUT, and so on.

IS_API

Type

Boolean

Description

True if this page was an API or Web Services request.

IS_ERROR

Type

Boolean

Description

True if this page was an error page.

IS_FIRST_REQUEST

Type

Boolean

Description	
1 if this page is the first Visualforce transaction in the request, or 0 if it isn't.	
IS_GUEST	Type Boolean
Description	
True if this page was a guest (unauthenticated) request.	
IS_SECURE	Type Boolean
Description	
True if this request is secure.	
LOGIN_KEY	Type Id
Description	
The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.	
Example	
GeJCsym5eyvtEK2I	
ORGANIZATION_ID	Type Id
Description	
The 15-character ID of the organization.	
Example	
00D000000000123	
PAGE_NAME	Type String
Description	
The name of the Visualforce page that was requested.	
QUERY	Type String
Description	
The SOQL query, if one was performed.	
REQUEST_ID	Type String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

REQUEST_STATUS

Type

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
- F: Failure
- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

REQUEST_TYPE

Type

String

Description

The request type.

Possible Values

- page: a normal request for a page
- content_UI: a content request for a page that originated in the user interface
- content_apex: a content request initiated by an Apex call
- PDF_UI: a request for a page in PDF format through the user interface
- PDF_apex: a request for PDF format by an Apex call (usually a Web Service call)

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

SITE_ID

Type

Id

Description

The 15-character ID of the Site.com site.

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

URI_ID_DERIVED

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

USER_ID

Type

Id

	Description The 15-character ID of the user who's using Salesforce services through the UI or the API.
	Example 00530000009M943
USER_ID_DERIVED	Type Id
	Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

	Example 005900000001SNIAO
--	-------------------------------------

Time-Based Workflow

Time-Based Workflow events contain details about queue activity monitoring.

Field	Details
DATA	Type String Description The record details of time queue activity.
EVENT_TYPE	Type String Description The type of event. Example ReportExport, URI, UITracking, API, RestApi, and so on.
LOG_GROUP_ID	Type String Description Marks log records that are committed or rolled back.
NUMBER_OF_RECORDS	Type Number Description The number of processed records.

ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
REQUEST_ID	Type String Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID. Example 3nWgxWbDKWWDIk0FKfF5DV
TIMESTAMP	Type String Description The access time of Salesforce services. Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ) Example 2015-07-27T11:32:59.555Z
TYPE	Type String Description The type of Apex callout. Example REST or AJAX

Transaction Security

Transaction Security events contain details about policy execution.

Field	Details
CLIENT_IP	Type IP
	Description The IP address of the client that's using Salesforce services.
	Example 96.43.144.26
CPU_TIME	Type Number
	Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
EVALUATION_TIME_MS	Type Number
	Description The time in milliseconds used to evaluate the policy.
EVENT_TIMESTAMP	Type String
	Description The time at which the Transaction Security event was generated in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ). This value may be earlier than TIMESTAMP_DERIVED by the amount of time taken to log the event.
	Example 2015-07-27T11:32:59.555Z
EVENT_TYPE	Type String
	Description The type of event.
	Example ReportExport, URI, UITracking, API, RestApi, and so on.
LOGIN_KEY	Type Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D00000000123

POLICY_ID

Type

Id

Description

The 15-character ID of the policy being evaluated.

Example

0053000009M943

POLICY_ID_DERIVED

Type

Id

Description

The 18-character case-insensitive ID of the policy being evaluated.

Example

005900000001SNIA0

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

RESULT

Type

String

Description

The outcome of evaluating the policy.

Example

TRIGGERED or NOT_TRIGGERED

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The time the event was logged in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ). This value may be later than EVENT_TIMESTAMP by the amount of time between when the event occurred and when it was logged.

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

	Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943
USER_ID_DERIVED	Type Id Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API. Example 00590000000I1SNIA0

UI Tracking

UI Tracking events contain details about user interactions with the mobile UI.

Field	Details
ACTION	Type EscapedString Description The hierarchical user action.
ACTION_LOCATION	Type String Description The name of the component where the user action occurred.
ACTION_TYPE	Type String Description The database action that's associated with the user action. Possible Values Database actions, including: <ul style="list-style-type: none"> • Create

- Update
- Delete
- Convert

APP_NAME	Type EscapedString Description The name of the application that the user accessed.
BROWSER_NAME	Type String Description The name of the browser that's accessed by the user.
BROWSER_VERSION	Type String Description The browser version that's accessed by the user.
CARRIER	Type String Description The name of the cellular service provider.
CLIENT	Type EscapedString Description The context of access. Example Phone, tablet, desktop, and so on.
CLIENT_ID	Type String Description The API client ID.
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26

CONNECTION_TYPE

Type

String

Description

The type of the connection.

Possible Values

- WIFI
- CDMA1x
- CDMA
- EDGE
- EVDO0
- EVDOA
- EVDOB
- GPRS
- HSDPA
- HSUPA
- HRPD
- LTE

DELTA

Type

Number

Description

The elapsed time between the start and end of the event in milliseconds.

DEVICE_ID

Type

String

Description

The unique device identifier.

EVENT_TYPE

Type

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOCATION

Type

EscapedString

	Description The hierarchical location.
NETWORK_ID	Type Id Description The network ID of the request.
NUMBER1	Type Number Description The record of a numeric value based on the operation. For example, the number of results returned by a search.
NUMBER2	Type Number Description The record of a numeric value based on operation, but for multiple values. Example NUMBER1 items selected in a list containing NUMBER2 items.
OBJECT_TYPE	Type String Description The API name of the record in the action.
ORGANIZATION_ID	Type Id Description The 15-character ID of the organization. Example 00D000000000123
OS_NAME	Type String Description The operating system name.
OS_VERSION	Type String Description The operating system version.

PAGE_OPTION

Type

String

Description

A comma-separated list of additional information about the current page.

Example

isVisualforce

RECORD_ID

Type

Id

Description

The ID of the record the user accessed.

RECORD_TYPE_ID

Type

Id

Description

The ID of the record type the user accessed.

REFERRER

Type

EscapedString

Description

The HTTP referrer header.

REQUEST_METHOD

Type

String

Description

The HTTP request method.

SDK_APP_NAME

Type

String

Description

The name of the mobile SDK application that's installed on the device.

Example

Salesforce1

SDK_APP_TYPE

Type

String

Description

The mobile SDK application type.

Example

Native, hybrid, and so on.

SDK_APP_VERSION

Type

String

Description

The mobile SDK application version.

Example

5.0

SDK_MODEL

Type

String

Description

The model of the mobile SDK application that's installed on the device.

Example

Salesforce1

SDK_VERSION

Type

String

Description

The mobile SDK version.

Example

2.1.0

SESSION_ID

Type

String

Description

The user's unique session ID.

SIGNAL_STRENGTH

Type

Number

Description

The cellular signal strength.

STATUS

Type

Boolean

Description

Indicates whether an error was logged. Information is logged in data.

TARGET

Type

EscapedString

Description

The request target.

TARGET2

Type
EscapedString

Description
Additional information for the request target.

TIMESTAMP

Type
String

Description
The access time of Salesforce services.

Example
20130715233322.670

TIMESTAMP_DERIVED

Type
Datetime

Description
The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example
2015-07-27T11:32:59.555Z

UNIQUE_PAGE_ID

Type
String

Description
The unique request ID for all the requests from 1 page.

USAGE_TIMESTAMP

Type
String

Description
The time when the usage occurred. Format: YYYYMMDDHHMM
SSsss.

USER_AGENT

Type
Escaped String

Description
The numeric code for the type of client used to make the request (for example, the browser, application, or API) as a string.

USER_ID

Type
Id

Description
The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED

Type

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

USER_TYPE

Type

String

Description

The user license of the user who's accessing Salesforce services through the UI or the API.

Possible Values

- S: Standard
- P: Partner
- p: Customer Portal Manager
- C: Customer Portal User
- O: Power Custom
- o: Custom
- L: Package License Manager
- X: Black Tab User
- N: Salesforce to Salesforce
- G: Guest
- D: External Who
- A: Automated Process
- b: High Volume Portal
- n: CSN Only
- F: Self Service

URI

URI events contain details about user interaction with the web browser UI.

Field	Details
CLIENT_IP	Type IP

Description

The IP address of the client that's using Salesforce services.

Example

96.43.144.26

CPU_TIME**Type**

Number

Description

The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.

DB_BLOCKS**Type**

Number

Description

Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.

DB_CPU_TIME**Type**

Number

Description

Allows you to monitor trends in database uptime.

DB_TOTAL_TIME**Type**

Number

Description

The time in nanoseconds for a database round trip. Compare this field to `CPU_TIME` to determine whether performance issues are occurring in the database layer or in your own code.

EVENT_TYPE**Type**

IP

Description

The IP address of the client that's using Salesforce services.

Example

96.43.144.26

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

REFERRER_URI

Type

String

Description

The referring URI of the page that's receiving the request.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDDlk0FKfF5DV

REQUEST_STATUS

Type

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
- F: Failure
- U: Undefined
- A: Authorization Error
- R: Redirect
- N: Not Found

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID

Type

Id

	Description The 15-character ID of the user who's using Salesforce services through the UI or the API.
	Example 00530000009M943
USER_ID_DERIVED	Type Id
	Description The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

	Example 005900000001SNIAO
--	-------------------------------------

Visualforce Request

VISUALFORCE__REQUEST__EVENT events contain details of Visualforce requests. Requests can originate from the browser (UI) or the API.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services.
	Example 96.43.144.26
CONTROLLER_TYPE	Type Number Description The type of controller that's used by the requested Visualforce page.
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
DB_BLOCKS	Type Number

Description

Indicates how much activity is occurring in the database. A high value for this field suggests that adding indexes or filters on your queries would benefit performance.

DB_CPU_TIME**Type**

Number

Description

Allows you to monitor trends in database uptime.

DB_TOTAL_TIME**Type**

Number

Description

The time in nanoseconds for a database round trip. Compare this field to `CPU_TIME` to determine whether performance issues are occurring in the database layer or in your own code.

EVENT_TYPE**Type**

String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

HTTP_METHOD**Type**

String

Description

The HTTP method of the request—GET, POST, PUT, and so on.

IS_AJAX_REQUEST**Type**

Boolean

Description

The value is `true` if the request is a partial page request.

IS_FIRST_REQUEST**Type**

Boolean

Description

1 if this page is the first Visualforce transaction in the request, or 0 if it isn't.

LOGIN_KEY**Type**

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

MANAGED_PACKAGE_NAMESPACE

Type

String

Description

If the page is part of a managed package, the namespace of that package.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

PAGE_NAME

Type

String

Description

The name of the Visualforce page that was requested.

QUERY

Type

String

Description

The SOQL query, if one was performed.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

REQUEST_SIZE

Type

Number

Description

The size of the request body, in bytes.

REQUEST_STATUS**Type**

String

Description

The status of the request for a page view or user interface action.

Possible Values

- S: Success
 - F: Failure
 - U: Undefined
 - A: Authorization Error
 - R: Redirect
 - N: Not Found
-

REQUEST_TYPE**Type**

String

Description

The request type.

Possible Values

- page: a normal request for a page
 - content_UI: a content request for a page that originated in the user interface
 - content_apex: a content request initiated by an Apex call
 - PDF_UI: a request for a page in PDF format through the user interface
 - PDF_apex: a request for PDF format by an Apex call (usually a Web Service call)
-

RESPONSE_SIZE**Type**

Number

Description

The size of the response, in bytes.

RUN_TIME**Type**

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY**Type**

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_AGENT

Type

Number

Description

The numeric code for the type of client used to make the request (for example, the browser, application, or API).

USER_ID

Type

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

0053000009M943

VIEW_STATE_SIZE

Type

Number

Description

The size of the Visualforce view state, in bytes.

Wave Change

Wave Change events represent route or page changes made in the Salesforce Wave Analytics user interface.

Field	Details
CLIENT_IP	Type IP
	Description The IP address of the client that's using Salesforce services.
	Example 96.43.144.26
CPU_TIME	Type Number
	Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
EVENT_TYPE	Type String
	Description The type of event.
	Example ReportExport, URI, UITracking, API, RestApi, and so on.
IS_NEW	Type Boolean

		Description	
		If the change routes to a new page, the value of this field is true. If it routes to an existing page, this field is false.	
LOGIN_KEY	Type Id	Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.	
		Example GeJCsym5eyvtEK2I	
ORGANIZATION_ID	Type Id	Description The 15-character ID of the organization.	
		Example 00D000000000123	
RECORD_ID	Type String	Description The Salesforce ID of the Wave object.	
REOPEN_COUNT	Type Number	Description If IS_NEW is false, the number of times that an existing page opens.	
REQUEST_ID	Type String	Description The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.	
		Example 3nWgxWbDKWWWDlk0FKfF5DV	
RUN_TIME	Type Number	Description The amount of time that the request took in milliseconds.	

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TIMESTAMP

Type

String

Description

The access time of Salesforce services.

Example

20130715233322.670

TIMESTAMP_DERIVED

Type

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example

2015-07-27T11:32:59.555Z

TYPE

Type

String

Description

The Wave object type.

URI

Type

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED

Type

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID	Type Id	Description The 15-character ID of the user who's using Salesforce services through the UI or the API.
	Example 00530000009M943	
WAVE_SESSION_ID	Type String	Description The ID of a particular session of Wave. Use this field to determine which log lines originated from a particular session.
WAVE_TIMESTAMP	Type Number	Description The time at which this log line was generated.

Wave Interaction

Wave Interaction events track user interactions with the Wave Analytics user interface.

Field	Details
CLIENT_IP	Type IP Description The IP address of the client that's using Salesforce services. Example 96.43.144.26
CPU_TIME	Type Number Description The CPU time in milliseconds used to complete the request. This field indicates the amount of activity taking place in the app server layer, highlighting pieces of Apex or Visualforce code that need refactoring.
EVENT_TYPE	Type String

Description

The type of event.

Example

ReportExport, URI, UITracking, API, RestApi, and so on.

LOGIN_KEY

Type

Id

Description

The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.

Example

GeJCsym5eyvtEK2I

NUM_CLICKS

Type

Number

Description

The number of clicks performed on a page in the Wave user interface.

NUM_SESSIONS

Type

Number

Description

The number of times a user returned to a particular page.

ORGANIZATION_ID

Type

Id

Description

The 15-character ID of the organization.

Example

00D000000000123

READ_TIME

Type

Number

Description

The amount of time a user spent on a particular tab.

RECORD_ID

Type

String

Description

The Salesforce ID of the Wave object.

REQUEST_ID**Type**

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example3nWgxWbDKWWDIk0FKfF5DV

RUN_TIME**Type**

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY**Type**

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Exampled7DEq/ANa7nNZZVD

TIMESTAMP**Type**

String

Description

The access time of Salesforce services.

Example20130715233322.670

TIMESTAMP_DERIVED**Type**

Datetime

Description

The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ)

Example2015-07-27T11:32:59.555Z

TOTAL_TIME**Type**

Number

Description

The total amount of time a tab was open in milliseconds.

TYPE**Type**

String

Description

The Wave object type.

URI**Type**

String

Description

The URI of the page that's receiving the request.

Example

/home/home.jsp

URI_ID_DERIVED**Type**

ID

Description

The 18-character case insensitive ID of the URI of the page that's receiving the request.

USER_ID**Type**

Id

Description

The 15-character ID of the user who's using Salesforce services through the UI or the API.

Example

00530000009M943

USER_ID_DERIVED**Type**

Id

Description

The 18-character case insensitive ID of the user who's using Salesforce services through the UI or the API.

Example

005900000001SNIA0

WAVE_SESSION_ID**Type**

String

Description

The ID of a particular session of Wave.

WAVE_TIMESTAMP**Type**

Number

Description

The time at which this log line was generated.

Wave Performance

Wave Performance events help you track trends in your Wave Analytics performance.

Field	Details
CLIENT_IP	<p>Type IP</p> <p>Description The IP address of the client that's using Salesforce services.</p> <p>Example 96.43.144.26</p>
EPT	<p>Type Number</p> <p>Description The experienced page time in milliseconds.</p>
EVENT_TYPE	<p>Type String</p> <p>Description The type of event.</p> <p>Example ReportExport, URI, UITracking, API, RestApi, and so on.</p>
LOGIN_KEY	<p>Type Id</p> <p>Description The ID that ties together all events in a given user's login session. This starts with a Login event and ends either with a Logout event or with the user session expiring.</p> <p>Example GeJCsym5eyvtEK2I</p>
NAME	<p>Type String</p> <p>Description The asset title or query string.</p>
ORGANIZATION_ID	<p>Type Id</p> <p>Description The 15-character ID of the organization.</p>

Example

00D000000000123

QUERY_ID

Type

String

Description

The ID of the Wave query.

RECORD_ID

Type

String

Description

The Salesforce ID of the Wave object.

REQUEST_ID

Type

String

Description

The unique ID of a single transaction. A transaction can contain one or more events. Each event in a given transaction has the same REQUEST_ID.

Example

3nWgxWbDKWWDIk0FKfF5DV

RUN_TIME

Type

Number

Description

The amount of time that the request took in milliseconds.

SESSION_KEY

Type

Id

Description

The ID that ties together all events for a particular activity session. For example, URI events while the user is interacting with an Accounts page.

Example

d7DEq/ANa7nNZZVD

TAB_ID

Type

String

Description

The ID of the particular Wave tab in the user interface.

TIMESTAMP

Type

String

	Description The access time of Salesforce services.
	Example 20130715233322.670
TIMESTAMP_DERIVED	Type Datetime Description The access time of Salesforce services in ISO8601-compatible format (YYYY-MM-DDTHH:MM:SS.sssZ) Example 2015-07-27T11:32:59.555Z
TYPE	Type String Description The Wave object type.
UI_RENDER_TIME	Type String Description The amount of time that it took for the user interface to render.
URI	Type String Description The URI of the page that's receiving the request. Example /home/home.jsp
URI_ID_DERIVED	Type ID Description The 18-character case insensitive ID of the URI of the page that's receiving the request.
USER_ID	Type Id Description The 15-character ID of the user who's using Salesforce services through the UI or the API. Example 00530000009M943

WAVE_SESSION_ID	Type String	
	Description The ID of a particular session of Wave.	
WAVE_TIMESTAMP	Type Number	
	Description The time at which this log line was generated.	

EventRelation

Represents people (a user, a lead, or contacts) or a resource (such as a conference room) invited to an event. This object lets you add or remove invitees from an event and use the API to manage invitees' responses to invitations. If Shared Activities is enabled, this object doesn't support triggers, workflow, or data validation rules.

EventRelation allows a variable number of relationships and handles deleted events differently, depending on whether Shared Activities is enabled.

If Shared Activities Is Enabled

- An event can be related to up to 50 contacts or one lead.
- If you delete an event, then relations between the event and any specified contacts, leads, and records are also deleted. The WhoId, WhatId, and AccountId field values are cleared. If you restore a deleted event, relations between the event and any specified contacts, leads, and records are also restored. The WhoId, WhatId, and AccountId field values are recalculated using the field values on EventRelation.

If Shared Activities Isn't Enabled

- An event can be related to only one contact or lead.
- If you delete an event, the WhoId, WhatId, and AccountId field values aren't cleared.

Whether or not Shared Activities is enabled, an event can be related to one other kind of record, such as an account, an opportunity, or a custom object (in addition to one or more contacts).



Note:

- With API versions 26.0 and later, the EventRelation object replaces the EventAttendee object, and the EventAttendee object is no longer visible. You can still query the EventAttendee object using packages that support API versions 25.0 and earlier, or by using Apex.
- An EventRelation object can't be created for a child event.
- EventRelation includes deactivated users.
- In API versions 25.0 and earlier, you can't use `query()`, `delete()`, or `update()` with events related to more than one contact.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `queryAll()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Contains the Account ID of the relation. For information on IDs, see ID Field Type. <code>AccountId</code> is visible when Shared Activities is enabled.</p>
EventId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Contains the ID of the event. This value can't be changed after it's been specified.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is <code>Deleted</code>.</p>
IsInvitee	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the relation is an invitee.</p> <ul style="list-style-type: none"> • <code>IsInvitee</code> is visible while Shared Activities is being enabled, after it has been enabled, and while it is being disabled. • <code>IsInvitee</code> defaults to <code>true</code> while Shared Activities is being enabled, after it has been enabled, and while it is being disabled if <code>IsInvitee</code>, <code>IsParent</code>, and <code>IsWhat</code> are not set. This configuration ensures compatibility when Shared Activities isn't enabled and EventRelation represents event invitees only. • <code>IsInvitee</code> defaults to <code>false</code> when Shared Activities is enabled if <code>IsParent</code> is set to <code>true</code>.
IsParent	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the relation is a contact or lead, or an account, opportunity, or other object. <code>IsParent</code> is visible when Shared Activities is enabled.</p>
IsWhat	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the relation is an account, an opportunity, or another object. Value is <code>false</code> if <code>RelationId</code> is a contact or a lead, and <code>true</code> otherwise. <code>IsWhat</code> is visible when Shared Activities is enabled.</p>
RelationId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Contains the ID of the person (User, Contact, or Lead) or the resource invited to an event. When Shared Activities is enabled, <code>RelationId</code> can also contain the ID of the contacts or lead related to an event. This value can't be changed after it's been specified.</p>
RespondedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Indicates the most recent date and time when the invitee responded to an invitation to an event.</p>
Response	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contains optional text that the invitee can enter when responding to an invitation to an event.</p>

Field	Details
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the invitee status with one of the following values:</p> <ul style="list-style-type: none"> • New: Invitee has received the invitation but hasn't yet responded. This value is the default. • Declined: Invitee has declined the invitation. • Accepted: Invitee has accepted the invitation. <p> Note: <i>Uninvited</i> and <i>Maybe</i> aren't currently supported.</p>

Usage

- Invitee related lists display slightly different content. In Salesforce1, the invitee related list includes invitees only, whereas in the full site, it also includes the event owner. To reproduce the full site functionality in Salesforce1, use the following API queries.

If you use Shared Activities in your Salesforce org, use this query:

```
SELECT RelationId FROM EventRelation WHERE IsInvitee = true AND eventId='[Event_Id]'
```

where *Event_Id* is the child event's ID.

If you don't use Shared Activities, use this query:

```
SELECT RelationId FROM EventRelation WHERE eventId='[Event_Id]'
```

These queries get the main event's relations and display them for the given child event. To further filter the results, add a `WHERE` clause.

Send email notifications

To send email notifications for a given event, query `EventRelation` for the event, iterate through the list, examine the status, and send email notifications to every person who accepted the invitation.

Determine what events a given invitee is attending

To determine all the events that a particular person is attending during a given time period (for example, next week), you can have a client application query the `Event` object for a given date range, iterate through the results, and, for each event, query the `EventRelation` object to determine whether the particular person (`RelationId`) has accepted an invitation to that event.

Create an invitee if Shared Activities is enabled (or during the process of enabling it or rolling back)

If the invitee is already a contact or lead, update `IsInvitee` to `true`.

If the invitee is not already a contact or lead, create an `EventRelation` object for the invitee with `IsInvitee` set to `true`.

Create an invitee if Shared Activities is not enabled

Create an `EventRelation` object for the invitee.

Query relations to a contact or a lead

```
List<EventRelation> whoRelations = [SELECT Id, Relation.Name FROM EventRelation WHERE EventId = '00UD000005zijD' AND isParent = true AND isWhat = false];
```

Query invitee relations

```
List<EventRelation> inviteeRelations = [SELECT Id, Relation.Name FROM EventRelation WHERE EventId = '00UD000005zijD' AND isInvitee = true];
```

Update an invitee relation to a contact or lead invitee relation

```
EventRelation er = [SELECT Id FROM EventRelation WHERE EventId = '00UD000005zijD' AND isInvitee = true and isParent = false LIMIT 1];
er.isParent = true;
update er;
```

Update a contact or lead relation to a contact or lead invitee relation

```
EventRelation er = [SELECT Id FROM EventRelation WHERE EventId = '00UD000005zijD' AND isParent = true and isInvitee = false LIMIT 1];
er.isInvitee = true;
update er;
```

Insert a contact or lead relation

```
EventRelation er = new EventRelation(EventId = '00UD000005zijH',
                                     RelationId = '003D00000Q8aeV', isParent = true, isInvitee = false);
insert er;
```

Insert an invitee relation

If `isParent`, `isWhat` and `isInvitee` are not set, and `RelationId` is a contact, lead, user, or calendar, `isInvitee` defaults to `true`. This means if an `EventRelation` isn't specifically inserted as a relation to a contact or lead, it's treated as an invitee relation by default.

```
EventRelation er = new EventRelation(EventId = '00UD000005zijH',
                                     RelationId = '003D00000Q8adv');
insert er;
```

Reproduce invitee related list functionality in Salesforce1

Invitee related lists display slightly different content in Salesforce1 and the full site. In Salesforce1, the invitee related list includes invitees only, whereas in the full site, it also includes the event owner.

If you use Shared Activities in your Salesforce org, use the following query to reproduce the full site functionality in Salesforce1:

```
SELECT RelationId FROM EventRelation WHERE isInvitee = true AND eventId='[Event_Id]'
```

where `Event_Id` is the child event's ID.

If you don't use Shared Activities, use this query:

```
SELECT RelationId FROM EventRelation WHERE eventId='[Event_Id]'
```

These queries get the main event's relations and display them for the given child event. To further filter the results, add a `WHERE` clause.

SEE ALSO:

[Event](#)
[EventWhoRelation](#)
[Object Basics](#)

EventTag

Associates a word or short phrase with an Event.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	Type reference Properties Create, Filter Description ID of the tagged item.
Name	Type string Properties Create, Filter Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.
TagDefinitionId	Type reference Properties Filter Description ID of the parent TagDefinition object that owns the tag.

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId.

Usage

EventTag stores the relationship between its parent TagDefinition and the Event being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

EventWhoRelation

Represents the relationship between an event and a lead or contacts. This derived object is a filtered version of the [EventRelation](#) on page 948 object; that is, `IsParent` is `true` and `IsWhat` is `false`. It doesn't represent relationships to invitees or to accounts, opportunities, or other objects. This object is available in API versions 29.0 and later.

EventWhoRelation allows a variable number of relationships: one lead or up to 50 contacts. Available only if you've enabled Shared Activities for your organization.

 **Note:** EventWhoRelation objects aren't created for child events.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
EventId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	<p>Description Indicates the ID of the event.</p>
RelationId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the contacts or lead related to the event.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates whether the person related to the event is a contact or lead.</p>

Usage

Apex example that queries relations to a contact or lead

```
List<EventWhoRelation> whoRelations = [SELECT Id, Relation.Name FROM EventWhoRelation WHERE EventId = '00UD0000005zijD'];
```

SEE ALSO:

[Event](#)
[EventRelation](#)

ExternalDataSource

Represents an external data source, which defines connection details for integration with data and content that are stored outside the Salesforce org. This object is available in API version 27.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
CustomConfiguration	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description A JSON-encoded configuration string that defines parameters specific to the type of external data source.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Endpoint	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description The URL of the external system, or if that URL is defined in a named credential, the named credential URL. A named credential URL contains the scheme <code>callout:</code>, the name of the named credential, and an optional path. For example: <code>callout:My_Named_Credential/some_path</code>. You can append a query string to a named credential URL. Use a question mark (?) as the separator between the named credential URL and the query string. For example: <code>callout:My_Named_Credential/some_path?format=json</code>.</p>
isWritable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Query, Sort</p>

Field Name	Details
	Description
	<p>Lets the Force.com platform and users in this org create, update, and delete records for external objects associated with the external data source. The external object data is stored outside the org. By default, external objects are read only.</p>
	<p>Available only for Salesforce Connect external data sources. Writable external objects aren't supported with the cross-org adapter for Salesforce Connect.</p>
Language	Type
	picklist
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	The language of the <code>MasterLabel</code> .
MasterLabel	Type
	string
	Properties
	Filter, Group, Sort
	Description
	Master label for the external data source. This internal label doesn't get translated.
NamespacePrefix	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p>
PrincipalType	Type
	picklist
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	<p>Specifies whether the org uses one set (<code>NamedUser</code>), multiple sets (<code>PerUser</code>), or no (<code>Anonymous</code>) credentials to access the external system. Each set of credentials corresponds to a login account on the external system. Corresponds to <code>Identity Type</code> in the user interface.</p>

Field Name	Details
Protocol	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Specifies whether to use OAuth, password authentication, or no authentication to access the external system. Some types of external data sources support only one value.</p> <ul style="list-style-type: none"> • For cloud-based Files Connect external systems, select Oauth 2.0. • For on-premises systems, select Password Authentication. • For Simple URL data sources, select No Authentication.
Repository	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Used for SharePoint Online. An optional name of the repository in the data source. Not applicable to all data source types.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Specifies the adapter that connects to the external system.</p>

Usage

Define an external data source to connect to data or content that's stored outside the Salesforce org. Then create external objects, which map to the external system's data and behave similarly to custom objects.

SEE ALSO:

[ExternalDataUserAuth](#)

[NamedCredential](#)

ExternalDataUserAuth

Stores authentication settings for a Salesforce user to access an external system. The external system must be defined in an external data source or a named credential that's configured to use per-user authentication. This object is available in API version 27.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ExternalDataSourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Salesforce ID of the external data source or named credential that defines the external system.</p>
Password	<p>Type encrypted string</p> <p>Properties Create, Nillable, Update</p> <p>Description Password portion of the credentials for the Salesforce user to access the external system.</p>
Protocol	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether to use OAuth, password authentication, or no authentication when the user accesses the external system.</p>
UserId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the Salesforce user who's accessing the external system.</p>
Username	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	Description
	Username portion of the credentials for the Salesforce user to access the external system.

Usage

These authentication settings enable a Salesforce user to access an external system. The external system is defined in Salesforce as one of the following.

- External data source—Provides access to external objects, whose data is stored outside the Salesforce organization.
- Named credential—Enables the user's actions to trigger authenticated callouts to the endpoint that's specified in the named credential.

If you grant users access to the external data source or named credential via permission sets or profiles, those users can manage their own authentication settings. See "Store Authentication Settings for External Systems" in the Salesforce Help.

SEE ALSO:

[ExternalDataSource](#)
[NamedCredential](#)

FeedAttachment

Represents an attachment to a feed item, such as a file attachment or a link. Use FeedAttachment to add various attachments to one feed item. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

- You can read, create, update, or delete a FeedAttachment only if you have the corresponding access to the associated FeedItem.
- Inline images aren't creatable, updatable, or deletable through SOAP API.

Fields

Field Name	Details
FeedEntityId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description
	<p>The ID of the associated feed entity that contains this attachment. Currently, the only feed entity supported is FeedItem.</p>
RecordId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort
	Description
	<p>The ID of the record that this feed attachment contains. For inline images, RecordId is a ContentVersion ID. For content attachments, RecordId is a ContentDocument ID.</p>
Title	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The title of this feed attachment. When Type is Link, Title value is the label for the attachment link. Otherwise, Title value isn't used.</p>
Type	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort
	Description
	<p>The type of this feed attachment. Valid values are:</p> <ul style="list-style-type: none"> • Content—A content attachment. • InlineImage—An inline image. The system creates an inline image attachment when an image is added to the body of the associated FeedItem. You can't add an inline image directly by using FeedAttachment. • Link—A link.
Value	Type
	string
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	<p>The string value of this FeedAttachment. This field is optional. If the feed attachment is a Link FeedAttachment, the value is the link URL string.</p>

Usage

- This Apex example shows how to add an attachment to a Lead using API version 36.0 and later. First, post a feed item.

```
//create and insert post
FeedItem post = new FeedItem();
post.Body = 'HelloThere';
post.ParentId = 'ID_OF_FEED_ITEM';
post.Title = 'FileName';
insert post;
```

Then insert the attachment.

```
//create and associate a content attachment to the post
FeedAttachment feedAttachment = new FeedAttachment();
feedAttachment.FeedEntityId = post.Id;
feedAttachment.RecordId = 'ID_OF_CONTENT_VERSION';
feedAttachment.Title = 'FileName';
feedAttachment.Type = 'CONTENT';
insert feedAttachment;
```

- You can only create one link attachment (FeedAttachment of type `Link`) per feed item.
- If the feed item type is one of the following, you can add content or link feed attachments to a FeedItem.
 - AdvancedTextPost
 - TextPost
 - ContentPost
 - LinkPost
 - QuestionPost
- When a FeedAttachment is added or removed from a feed item, Salesforce updates the type of the feed item to its most appropriate value, as follows.
 - If all content feed attachments are removed from a feed item of type `ContentPost`, the type of this feed item is updated to `TextPost`.
 - Conversely, if a content feed attachment is added to a feed item of type `TextPost`, the type of this feed item is updated to `ContentPost`.
 - If all link feed attachments are removed from a feed item of type `LinkPost`, the type of this feed item is updated to `TextPost`.
 - Conversely, if a link feed attachment is added to a feed item of type `TextPost`, the type of this feed item is updated to `LinkPost`.
 - The type of all other feed items, such as `QuestionPost` or `AdvancedTextPost` feed items, doesn't change when any feed attachments are added or removed.
 - If a content feed attachment is added to a feed item of type `LinkPost`, the feed item type is updated to `ContentPost`.
 - If all content attachments are removed from a feed item of type `ContentPost`, but there's also a link attachment, the feed item type is updated to `LinkPost`.
- Those without administrator privileges can't retrieve a FeedAttachment by its ID in a SOQL query. They can retrieve attachments by specifying the associated `FeedEntityId`, as follows:

```
SELECT Id FROM FeedAttachment WHERE FeedEntityId = 'some_feedItem_id'
```

- Alternatively, retrieve attachments by using a SOQL query on FeedItem with a subquery on the FeedAttachments child relationship, as follows.

```
SELECT Body, (SELECT RecordId, Title, Type, Value FROM FeedAttachments)
FROM FeedItem
WHERE Id = 'some_feedItem_id'
```

- FeedAttachment is not a triggerable object. You can access feed attachments in FeedItem *update* triggers by retrieving them through a SOQL query. For a trigger example, and to learn about trigger considerations for FeedAttachment, see [Triggers for Chatter Objects](#) in the *Apex Developer Guide*.

FeedComment

Represents a comment added to a feed by a user. This object is available in API version 18.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Note the following when working with feed comments.

- You must have read access to the feed's parent type to see a FeedComment record.
- You must have access to the feed to add a comment.
- If the comment is related to a user record, the user can delete the comment. For example, if John Smith makes a comment about Sasha Jones, Sasha can delete the comment.
- If the logged-in user has the "Insert System Field Values for Chatter Feeds" user permission, the `create` field property is available on `CreatedBy` and `CreatedDate` system fields. During migration, the logged-in user can set these fields to the original post's author and creation date. The fields can't be updated after migration.

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the object associated with the feed and delete permission on the parent feed
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

- "Manage Unlisted Groups"

Only users with this permission can delete items in unlisted groups.

Fields

Field	Details
CommentBody	Type
	textarea

Field	Details
	<p>Properties Create, Filter, Sort, Update</p> <p>Description The text in the comment.</p>
CommentType	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of comment:<ul style="list-style-type: none">• ContentComment—an uploaded file on a comment• TextComment—a direct text entry on a commentBefore API version 24.0, a text entry was required on a comment. As of version 24.0, a text entry is optional if the CommentType is ContentComment.</p>
FeedItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the feed containing the comment.</p>
InsertedById	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the InsertedBy value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the feed CommentBody contains rich text. Set IsRichText to true if you post a rich text feed comment using the SOAP API. Otherwise, the comment is rendered as plain text. Rich text supports the following HTML tags:</p>

Field	Details
	<ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only through the API and must reference files in Salesforce similar to this example: </p> <p> Note: This attribute is available as of API version 38.0. In API version 38.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 37.0 and prior, all rich text appears as a plain-text representation.</p>
LastEditBy.Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the user who last edited the feed comment.</p>
LastEditDate	<p>Type datetime</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description The date the feed comment was last edited.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of a record associated with the feed comment. For example, if you are commenting on a change to a field on Account, ParentId is set to the account ID.</p>

Field	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentComment. This field is null for all comments except ContentComment.</p> <p>For example, set this field to an existing ContentVersion and post it to a comment as a FeedComment object of CommentTypeContentComment.</p>
Revision	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The number of times the comment was revised.</p>
Status	<p>Type picklist</p> <p>Properties Create, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed comment is published and visible to all who have access to the parent feed item. To make changes to a comment's status, the comment's parent feed item must be in a published state. This field is available in API version 38.0 and later.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Published—The comment is visible to all who have access to the parent feed item. • PendingReview—The comment is visible to its author and to users who can see the parent feed item and have "ViewAllData" or "CanApproveFeedPostAndComment" permission. The comment can be deleted by its author and users who can see the comment and have "CanApproveFeedPostAndComment" or "ModifyAllData" permission. If the parent feed item is published, the comment can be edited by its author and users who can see the comment and have "CanApproveFeedPostAndComment" or "ModifyAllData" permission. Comment status can be changed from Published to PendingReview and from PendingReview to Published by users who have "CanApproveFeedPostAndComment" or "ModifyAllData" permission. <p>Some actions are blocked when a feed comment is pending review:</p> <ul style="list-style-type: none"> — Select as Best—When a feed comment that is marked as best answer becomes unpublished, it's removed as the best answer. If the comment is published, its best answer status is not restored. — Like and unlike

Field	Details
SystemModstamp	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter</p> <p>Description Date and time when a user or automated process (such as a trigger) last modified this record. SystemModstamp is a read-only system field, available in FeedComment as of API version 37.0.</p>

Usage

- As of API version 23.0 and later, if you have “View All Data” permission, you can query FeedComment records directly without an ID filter. If you don’t have “View All Data” permission, you can’t query FeedComment records directly, with or without an ID filter.

For example, the following query returns general information about a feed:

```
SELECT ID, CreatedDate, CreatedById, CreatedBy.FirstName,
       CreatedBy.LastName, ParentId, Parent.Name, Body
FROM FeedItem
WHERE CreatedDate > LAST_MONTH
ORDER BY CreatedDate DESC, Id DESC
```

- You can search for text in comments using SOSL. For example, the following Java class uses `search()` to find the string “foo” in any field of a record:

```
public void searchSample() {
    try {
        SearchResult sr = connection.search("find {foo} in all fields " +
            "returning feedcomment(Id, FeedItemId, CommentBody)");
        // Put the results into an array of SearchRecords
        SearchRecord[] records = sr.getSearchRecords();
        // Check the length of the returned array of records to see
        // if the search found anything
        if (records != null && records.length > 0) {
            System.out.println("Found " + records.length + " comments: ");
            // Display each comment
            for (SearchRecord record : records) {
                FeedComment comment = (FeedComment) record.getRecord();
                System.out.println(comment.getId() + ":" +
                    comment.getCommentBody());
            }
        } else {
            System.out.println("No records were found for the search.");
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

- If you use an Apex trigger to modify the `Body` of a FeedComment object, all mentions hyperlinks are converted to plain text. The mentioned users don't get email notifications.

 **Note:** This object is hard deleted. It isn't sent to the Recycle Bin.

SEE ALSO:

[NewsFeed](#)

[UserProfileFeed](#)

FeedItem

FeedItem represents an entry in the feed, such as changes in a record feed, including text posts, link posts, and content posts. This object is available in API version 21.0 and later. This object replaces FeedPost.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`,
`update()`, `upsert()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"

 **Note:** Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

- "Manage Unlisted Groups"

Only users with this permission can delete items in unlisted groups.

Only users with the "Modify All Data" permission can delete a FeedItem of Type `TrackedChange`.

If the logged-in user has the "Insert System Field Values for Chatter Feeds" user permission, the `create` field property is available on `CreatedBy` and `CreatedDate` system fields. During migration, the logged-in user can set these fields to the original post's author and creation date. The fields can't be updated after migration.

Fields

Field Name	Details
<code>BestCommentId</code>	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The Id of the comment marked as best answer on a question post.</p>

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Create, Nillable, Sort, Update on page 71</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p> <p> Tip: See the IsRichText field for a list of HTML tags supported in the body of rich text posts.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Create, Nillable</p>

Field Name	Details
	Description
	Available in API version 35.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code> . Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code> .
ContentDescription	Type textarea
	Properties
	Create , Nillable , Sort
	Description
	Available in API version 35.0 and earlier only. The description of the file specified in <code>ContentData</code> .
ContentFileName	Type string
	Properties
	Create , Group , Nillable , Sort
	Description
	Available in API version 35.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code> . The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code> .
ContentSize	Type int
	Properties
	Group , Nillable , Sort
	Description
	Available in API version 35.0 and earlier only. This field is the size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.
ContentType	Type string
	Properties
	Group , Nillable , Sort
	Description
	Available in API version 35.0 and earlier only. This field is the MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.
FeedPostId	Type reference

Field Name	Details
	Properties
	Filter , Group , Nillable , Sort
	Description
	<p>This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only.</p>
	<p>ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
HasContent	Type
	boolean
	Properties
	Defaulted on create , Filter , Group , Sort
	Description
	<p>Indicates whether the feed item has content.</p>
HasLink	Type
	boolean
	Properties
	Defaulted on create , Filter , Group , Sort
	Description
	<p>Indicates whether the feed item has a link attached.</p>
InsertedById	Type
	reference
	Properties
	Filter , Group , Nillable , Sort
	Description
	<p>ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	Type
	boolean
	Properties
	Defaulted on create , Filter , Group , Sort
	Description
	<p>Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	Type
	boolean

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description</p> <p>Indicates whether the feed item Body contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastEditById	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description</p> <p>ID of the user who last edited the feed item.</p>
LastEditDate	<p>Type datetime</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description</p> <p>The date the feed item was last edited.</p>

Field Name	Details
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Create, Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p>
	<p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities.
	<p>Note the following exceptions for NetworkScope:</p>
	<ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the object type to which the FeedItem object is related. For example, set this field to a UserId to post to someone's profile feed, or an AccountId to post to a specific account.</p>

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. For Work.com thanks posts, it's the ID of the WorkThanks object associated with a RypplePost. This field is typically null for all posts except ContentPost and RypplePost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Revision	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The revision number of the feed item.</p>
Status	<p>Type picklist</p> <p>Properties Create, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is published and visible to all who have access to the feed. This field is available in API version 37.0 and later.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Published—The feed item is visible to all who have access to the feed. • PendingReview—The feed item is visible to its author and to users who can see the item and have "ViewAllData" or "CanApproveFeedPostAndComment" permission. The feed item can be deleted and edited by its author and users who can see the item and have "CanApproveFeedPostAndComment" or "ModifyAllData" permission. <p>Some actions are blocked when a feed item is pending review:</p> <ul style="list-style-type: none"> – Comment – Like/unlike – Bookmark – Share
Title	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Group, Nillable, Sort, Update</p>
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name. The Title field can be updated on posts of Type QuestionPost.</p>
Type	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description The type of FeedItem. Except for ContentPost, LinkPost, and TextPost, don't create FeedItem types directly from the API.</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p>
	<ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile.

Field Name	Details
	<ul style="list-style-type: none"> • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • AttachArticleEvent—generated event when a user attaches an article to a case. • CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • CaseCommentPost—generated event when a user adds a case comment for a case object. • ChangeStatusPost—generated event when a user changes the status of a case. • ChatTranscriptPost—generated event when Live Agent transcript is saved to a case. • EmailMessageEvent—generated event when an email related to a case object is sent or received. • FacebookPost—generated when a Facebook post is created from a case. Deprecated. • MilestoneEvent—generated when a case milestone is completed or reaches violation status. • SocialPost—generated when a social post is created from a case.
	 Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none"> • AllUsers—The feed item is available to all users who have permission to see the feed item.

Field Name	Details
	<ul style="list-style-type: none"> InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none"> For record posts, Visibility is set to InternalUsers for all internal users by default. External users can set Visibility only to AllUsers. Visibility can be updated on record posts. The Update property is supported only for feed items posted on records.

Usage

- This Apex example shows how to add an attachment to a Lead using API version 36.0 and later. First, post a feed item.

```
//create and insert post
FeedItem post = new FeedItem();
post.Body = 'HelloThere';
post.ParentId = 'ID_OF_LEAD_ENTITY';
post.Title = 'FileName';
insert post;
```

Then insert the attachment.

```
//create and associate a content attachment to the post
FeedAttachment feedAttachment = new FeedAttachment();
feedAttachment.FeedEntityId = post.Id;
feedAttachment.RecordId = 'ID_OF_CONTENT_VERSION';
feedAttachment.Title = 'FileName';
feedAttachment.Type = 'CONTENT';
insert feedAttachment;
```

- If you are using API version 23.0 or later and have “View All Data” permission, you can directly query for a FeedItem. The following example returns the 20 most recent feed items.

```
SELECT ID, CreatedDate, CreatedById, CreatedBy.FirstName, CreatedBy.LastName, ParentId,
Parent.Name, Body,
(SELECT ID, FieldName, OldValue, NewValue FROM FeedTrackedChanges ORDER BY ID DESC)
FROM FeedItem
WHERE CreatedDate > LAST_MONTH
ORDER BY CreatedDate DESC
```

- If you are using an earlier API version than version 23.0, query FeedItem objects through a feed (such as [AccountFeed](#) or [OpportunityFeed](#)). The following example returns all feed items for a given account, ordered by date descending:

```
SELECT Id, Type, FeedItem.Body
FROM AccountFeed
WHERE ParentId = AccountId ORDER BY CreatedDate DESC
```



Note: Provide the ParentId for API version 22.0 and earlier.

- A FeedItem of type `UserStatus` is automatically created when a user adds a post to update the status. You can't explicitly create a FeedItem of type `UserStatus`.
- The FeedItem object doesn't support aggregate functions in queries.
- If the logged-in user has the "Insert System Field Values for Chatter Feeds" user permission, the `create` field property is available on `CreatedBy` and `CreatedDate` system fields. During migration, the logged-in user can set these fields to the original post's author and creation date. The fields can't be updated after migration.
- The size limit for an attachment on a profile or news feed is 25 MB.
- The size limit for an attachment on a record feed is 5 MB.
- You can't use the content fields to update or delete the content.
- You can't filter or update the content fields.
- Deleting a FeedItem via the API also deletes the associated content. Likewise, undeleting a FeedItem restores associated content.



Note: This object is hard deleted. It isn't sent to the Recycle Bin.

- After uploading to a feed, it is possible for an attachment or document to be deleted, marked private, or hidden by sharing rules. In this case, all content fields in a FeedItem object appear as `null` in a SOQL query.
- You can't explicitly create or delete a `FeedTrackedChange` record.
- If you insert a FeedItem or `FeedComment` of Type `ContentPost` on a `User` or `Group` to create a file, the `NetworkScope` field value of the FeedItem is passed to the file.
- If you use an Apex trigger to modify the `Body` of a FeedItem object, all mentions hyperlinks are converted to plain text. The mentioned users don't get email notifications.
- If you insert rich text into the feed item body, make sure that the case of the opening and closing HTML tags matches. For example, `This is bold text` generates an error.
- To check file sharing with Apex triggers, write triggers on ContentDocumentLink instead of FeedItem. For a ContentDocumentLink trigger example, see [ContentDocumentLink](#).
- In API version 36.0 and later, use `FeedAttachment` to attach one or more content items to a FeedItem. As a result of multiple attachment support through FeedAttachment, all fields related to content attachments have been removed. These fields are: `ContentData`, `ContentDescription`, `ContentFileName`, `ContentSize`, and `ContentType`.

FeedLike

Indicates that a user has liked a feed item. This object is available in API version 21.0 and later.

FeedLike records represent likes on posts and not likes on comments. Likes on comments can't be queried via the API. A FeedLike is a child object of an associated FeedItem, FeedTrackedChange, or object feed, such as AccountFeed.

Supported Calls

`create()`, `delete()`, `describeSObjects()`

Special Access Rules

If the logged-in user has the "Insert System Field Values for Chatter Feeds" user permission, the `create` field property is available on `CreatedBy` and `CreatedDate` system fields. During migration, the logged-in user can set these fields to the original post's author and creation date. The fields can't be updated after migration.

Fields

Field Name	Details
FeedItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the feed item that the user liked.</p>
FeedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The Id of a feed item or feed comment the user liked. If the user liked a comment, FeedEntityId is set to the ID of the comment. If the user liked a feed item, FeedEntityId is set to the ID of the feed item. FeedEntityId is an optional field. The default value is the ID of the feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the InsertedBy value is set to the ID of the logged-in user.</p>

Usage

You can't query FeedLike records directly. They can only be queried via the entity feed, such as AccountFeed.

FeedLike records represent likes on posts and not likes on comments. Likes on comments can't be queried via the API.

FeedPollChoice

Shows the choices for a poll posted in the feed. This object is available in API version 29.0 and later.

Supported Calls

`query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

- "Manage Unlisted Groups"

Only users with this permission can delete items in unlisted groups.

Fields

Field Name	Details
ChoiceBody	Type string Properties Group Description A choice in the poll.
FeedItemId	Type reference Properties Create, Filter, Group, Sort Description ID of the feed item for the poll.
Position	Type int Properties Group, Sort Description Shows the position of the poll choice.

Usage

Use this object to query all of the choices associated with a particular poll. To view how people voted on the poll, see the FeedPollVote object.

FeedPollVote

Shows how users voted on a poll posted in the feed. This object is available in API version 29.0 and later.

Supported Calls

`query()`, `retrieve()`

Fields

Field Name	Details
ChoiceId	Type reference Properties Filter, Group Description Indicates which choice a user selected on a poll posted in a feed.
FeedItemId	Type reference Properties Filter, Group, Sort Description ID of the feed item for the poll.

Usage

Use this object to query how users voted on a particular poll.

FeedPost

FeedPost represents the following types of changes in a record feed, such as AccountFeed: text posts, link posts, and content posts. This object is available in API version 18.0 through 21.0. FeedPost is no longer available in later versions. Starting with API version 21.0, use FeedItem to represent text posts, link posts, and content posts in feeds.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `search()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Only users with the "Modify All Data" permission can delete a FeedItem of `Type TrackedChange`.

If the logged-in user has the "Insert System Field Values for Chatter Feeds" user permission, the `CreatedBy` and `CreatedDate` system fields. During migration, the logged-in user can set these fields to the original post's author and creation date. The fields can't be updated after migration.

Fields

Field	Details
Body	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description The content of the FeedPost. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
ContentData	<p>Type base64</p> <p>Properties Create, Nillable</p> <p>Description This field is required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type string</p> <p>Properties Create, Nillable, Sort</p> <p>Description The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Create, Group, Nillable, Sort</p>

Field	Details
	<p>Description This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nullable, Sort</p> <p>Description This field is the size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nullable, Sort</p> <p>Description This field is the MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
FeedItemId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the feed containing the FeedPost.</p>
InsertedById	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the InsertedBy value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). This field is a standard system field.</p>
LinkUrl	<p>Type <code>url</code></p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
ParentId	<p>Type <code>reference</code></p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the object type to which the <code>FeedPost</code> is related. For example, set this field to a <code>UserId</code> to post to someone's profile feed, or an <code>AccountId</code> to post to a specific account.</p>
Title	<p>Type <code>string</code></p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The title of the <code>FeedPost</code>. When the <code>Type</code> is <code>LinkPost</code>, the <code>Body</code> is the URL and the <code>Title</code> is the label for the link.</p>
Type	<p>Type <code>picklist</code></p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The type of <code>FeedPost</code>:</p> <ul style="list-style-type: none"> • <code>UserStatus</code>—automatically generated when a user updates their status • <code>TrackedChange</code>—ignore • <code>TextPost</code>—a direct text entry on a feed • <code>LinkPost</code>—a URL posting on a feed • <code>ContentPost</code>—an uploaded file on a feed <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, you must specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Usage

- You can't directly query for a FeedPost. FeedPosts are always associated with a feed item, so you can query for them through the feeds. The following example returns all feed items for a given account, ordered by date descending:

```
SELECT Id, Type, FeedPost.Body  
FROM AccountFeed  
WHERE ParentId = AccountId ORDER BY CreatedDate DESC
```

- A FeedPost of type UserStatus is automatically created when a user adds a post to update the current status. You can't explicitly create a FeedPost of type UserStatus.
- The size limit for an attachment on a profile, news, or record feed is 2 GB.
- You can't use the content fields to update or delete the content.
- You can't filter or update the content fields.
- Deleting a FeedPost via the API also deletes the associated content and FeedPost objects. Likewise, undeleting a FeedPost restores associated content and FeedPost objects.



Note: This object is hard deleted. It isn't sent to the Recycle Bin.

- After uploading to a feed, it is possible for an attachment or document to be deleted, marked private, or hidden by sharing rules. In this case, all content fields in FeedPost appear as null in a SOQL query.
- You can't explicitly create or delete a FeedTrackedChange record.

FeedRevision

Holds the revision history of a specific feed item or comment, including a list of attributes that changed for each revision. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
Action	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Holds the type of modification to the underlying feed item or comment attribute. Action can have the value Changed.</p>
EditedAttribute	<p>Type picklist</p>

Field Name	Details
	Properties Filter , Group , Nillable , Sort
	Description
	Identifies the part of the feed item or comment which was modified. A single revision can have many edited attributes.
FeedEntityId	Type reference
	Properties Filter , Group , Sort
	Description
	Identifies the modified feed item or comment.
IsDeleted	Type boolean
	Properties Defaulted on create , Filter , Group , Sort
	Description
	Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). This field is a standard system field.
IsValueRichText	Type boolean
	Properties Defaulted on create , Filter , Group , Sort
	Description
	Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.
	Rich text supports the following HTML tags:
	<ul style="list-style-type: none"> • <code><p></code>
	 Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.
	<ul style="list-style-type: none"> • <code></code>
	<ul style="list-style-type: none"> • <code><code></code>
	<ul style="list-style-type: none"> • <code><i></code>
	<ul style="list-style-type: none"> • <code><u></code>
	<ul style="list-style-type: none"> • <code><s></code>
	<ul style="list-style-type: none"> • <code></code>
	<ul style="list-style-type: none"> • <code></code>

Field Name	Details
	<ul style="list-style-type: none"> • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
OriginNetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the community in which a user modified the feed item or comment. This field is only available, if Salesforce Communities is enabled for your organization.</p>
Revision	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The revision number of the feed item or comment.</p>
Value	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description Identifies the value of the <code>EditedAttribute</code> field before the update.</p>

Usage

This object tracks the changes made to a feed item or feed comment and stores a list of attributes that changed for each revision.

- To query the FeedRevision object, users need the "View All Data" permission or supply a WHERE clause on the `FeedEntityId`.

FeedTrackedChange

Represents an individual field change or set of field changes. A FeedTrackedChange is a child object of a record feed, such as AccountFeed. This object is available in API version 18.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Defaulted on create, Group, Restricted picklist, Sort</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO currency code for the field, if <code>FieldName</code> is a currency field.</p>
FeedItemId	<p>Type reference</p> <p>Properties Group, Sort, Filter</p> <p>Description ID of the parent feed that tracks the field change.</p>
FieldName	<p>Type string</p> <p>Properties Group, Sort</p> <p>Description The name of the field that was changed.</p> <p> Note: This field also tracks other events that are not related to an individual field for a parent feed. These events occur as the parent record advances through its pipeline. For example, a value of <code>leadConverted</code> indicates that a lead has been converted to an opportunity. For a full list of values, see Tracking of Special Events.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field	Details
	<p>Description The new value of the field that was changed.</p>
OldCurrencyIsoCode	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO currency code for the OldValue field, if <code>FieldName</code> is a currency field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The last value of the field before it was changed.</p>

Usage

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later.

If you move a custom field to the Recycle Bin, all FeedTrackedChange records that track historical changes to the custom field are automatically deleted and are not restored if the custom-field is undeleted.

The following sections outline the difference between standard feeds and custom feeds.

Standard Feeds

A standard feed is a record feed, such as AccountFeed. FeedTrackedChange records for standard feeds can only be queried via the parent feed object.

Note the following when working with standard feed items:

- Feed items for standard feeds are read only in the API.
- A FeedTrackedChange record is visible when you have read access on the record feed, and when the field in the FeedTrackedChange is visible in the field-level security settings.

Custom Feeds

If you want more control over the information provided in a record feed, such as AccountFeed, you can create a custom feed. A custom feed can replace or augment an existing record feed. For example, you might want to:

- Disable the standard account record feed and use an Apex trigger to generate FeedTrackedChange records for the events that you want to track in the feed instead.
- Augment the standard contact record feed by writing an API client that inserts feed items for events that are not tracked in the standard feed.

Tracking of Special Events

The `FieldName` field also tracks other events that are not related to an individual field for a parent feed. These events occur as the parent record advances through its pipeline. For example, a value of `leadConverted` indicates that a lead has been converted to an opportunity.

Valid values for the `FieldName` field for multiple objects:

- `created`
- `ownerAccepted`
- `ownerAssignment`

Additional valid values for the `FieldName` field for individual objects:

Account

- `accountCreatedFromLead`
- `accountMerged`
- `accountUpdatedByLead`
- `personAccountUpdatedByLead`

Case

- `closed`
- `ownerEscalated`

Contact

- `contactCreatedFromLead`
- `contactMerged`
- `contactUpdatedByLead`

Contract

- `contractActivation`
- `contractApproval`
- `contractConversion`
- `contractExpiration`
- `contractTermination`

Lead

- `leadConverted`
- `leadMerged`

Opportunity

- opportunityCreatedFromLead

SEE ALSO:

[NewsFeed](#)

[UserProfileFeed](#)

FieldHistoryArchive

Represents field history values for all objects that retain field history. This is a BigObject, available only to users with the “Retain Field History” permission. This object is available in API version 29.0 and later.

Each instance of the `FieldHistoryArchive` object represents a single change in the value of a field. `FieldHistoryArchive` stores history for both standard and custom fields.

The `Field` field returns the name of the field unless the parent field or object is deleted, in which case it returns the field ID. You can use the ID to retrieve the old field and object name from the `FieldNameAfterArchival` and `ParentNameAfterArchival` fields, respectively.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field Name	Details
ArchiveFieldName	<p>Type string</p> <p>Properties Nillable</p> <p>Description The name of the field at the time the data was archived. If the field name changed, the name is sometimes not the same for all records related to a single field.</p>
ArchiveParentName	<p>Type string</p> <p>Properties Nillable</p> <p>Description The name of the parent object at the time the data was archived. If the object name changed, the name is sometimes not the same for all records related to a single field.</p>

Field Name	Details
ArchiveParentType	<p>Type string</p> <p>Properties Nillable</p> <p>Description The type of the field at the time the data was archived. If the field type changed, the type is sometimes not the same for all records related to a single field.</p>
ArchiveTimestamp	<p>Type dateTime</p> <p>Properties Nillable</p> <p>Description The date and time at which the data was archived.</p>
CreatedBy	<p>Type dateTime</p> <p>Properties Nillable</p> <p>Description The user ID of the user who created the original record.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Nillable, Sort</p> <p>Description The date and time at which the original record was created.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist</p> <p>Description The name of the field that was changed. If the field is deleted from the parent object, the Field field contains the field ID instead.</p>
FieldHistoryType	<p>Type picklist</p> <p>Properties Nillable, Sort, Restricted picklist</p>

Field Name	Details
	<p>Description The name of the object that contains the field history (for example, Account).</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, idLookup</p> <p>Description The ID of the archived record. It's useful to have a field's ID for fields that you've deleted. (Field names aren't retained in history when you delete fields from Salesforce.)</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description The new value of the modified field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable</p> <p>Description The previous value of the modified field.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The ID of the object that contains the field (the parent object).</p>

Usage

When sorting fields, order them as follows:

1. FieldHistoryType ASC
2. ParentID ASC
3. CreatedDate DESC

FieldPermissions

Represents the enabled field permissions for the parent PermissionSet. This object is available in API version 24.0 and later.

To grant a user access to a field, associate a FieldPermissions record with a PermissionSet that's assigned to a user. FieldPermissions records are only supported in PermissionSet, not in Profile.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The field's API name. This name must be prefixed with the <code>SObjectType</code>. For example, <code>Merchandise__c.Description__c</code></p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The <code>Id</code> of the field's parent PermissionSet.</p>
PermissionsEdit	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, users assigned to the parent PermissionSet can edit this field. Requires <code>PermissionsRead</code> for the same field to be <code>true</code>.</p>
PermissionsRead	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description If true, users assigned to the parent PermissionSet can view this field. A FieldPermissions record must have at minimum PermissionsRead set to true, or it will be deleted.</p>
SObjectType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The object's API name. For example, Merchandise__c.</p>

Usage

FieldPermissions work similarly to ObjectPermissions. However, FieldPermissions includes a `Field` attribute to return the name of the field.

For example, the following query returns all FieldPermissions records that have at least the "Read" permission. The results includes the field, object, and permission set names.

```
SELECT SObjectType, Field, PermissionsRead, Parent.Name
FROM FieldPermissions
WHERE PermissionsRead = True
```

Include the field's parent object when querying FieldPermissions. For example, to find all rows that match the Account object's `Type` field, create the following query:

```
SELECT Id, SObjectType, Field
FROM FieldPermissions
WHERE Field = 'Account.Type' AND SObjectType = 'Account'
```

Both `SObjectType` and `Field` must be included in the `SELECT` line of the query. You must also provide the full API name of the field in the form of `SObjectType.Field` when querying for a field.

Special Properties for Field Permissions

The auto-number and formula fields have special rules for how field permissions work. Both have FieldPermissions records, but inserting and updating is limited to `PermissionsRead`. `PermissionsEdit` isn't allowed for either field type, since these fields must be read-only for users.

The following field types don't return a FieldPermissions record because they are assumed to always be readable.

- `Id`
- `CreatedById`
- `CreatedDate`
- `IsDeleted`
- `LastModifiedById`
- `LastModifiedDate`

- SystemModStamp

The following field types don't return a FieldPermissions record because they are assumed to always be readable and writable.

- OwnerId
- Master-detail custom (relationship) fields
- Universally required custom fields

As a result, the following query returns no records, even though users do have some access to some of the fields.

```
SELECT Field, SObjectType, PermissionsRead
FROM FieldPermissions
WHERE Field='Id'
```

To determine if a field can return a FieldPermissions record, you can call a `describeSObject()` on the field. For example, `describeSObject('Merchandise__c')`, returns all the properties of the Merchandise custom object, including field properties. If you're using a field whose `permissionable` property is `false` (such as any of the field types listed in this section), you can't query, insert, update, or delete any field permissions records, because none exist.

Working with Custom Activity Fields

While tasks and events are considered separate objects, they share a common set of activity custom fields. As a result, when a custom task field is created, a custom event field is also created, and vice versa. You can display the custom field on the event layout, task layout, or both event and task layouts.

Although custom activity fields are shared between tasks and events, you'll see separate FieldPermissions records for the task and event. However, changes made to one field permission record are automatically made to the other. For example, if you create a custom activity field, assign field permissions to it in a permission set, and run the following query, the query will return two records with the same permission value.

```
SELECT Field, Id, ParentId, PermissionsEdit, PermissionsRead, SObjectType
FROM FieldPermissions
WHERE SObjectType = 'event' OR SObjectType = 'task'
```

If you then update one of the records with a different set of field permission values and run the query again, the same permission values for both records are returned.

Nesting Field Permissions

You can nest FieldPermissions in a PermissionSet query. For example, the following returns any permission sets where "Edit Read Only Fields" is `true`. Additionally, the result set will include both the "Read" and "Edit" field permission on the Merchandise object. This is done by nesting the SOQL with a field permission query using the relationship name for field permissions: `FieldPerms`.

```
SELECT PermissionsEditReadonlyFields,
(SELECT SObjectType, Field, PermissionsRead, PermissionsEdit
FROM FieldPerms
WHERE SObjectType = 'Merchandise__c')
FROM PermissionSet
WHERE PermissionsEditReadonlyFields = true
```

As a result, it's possible to traverse the relationship between the PermissionSet and any child related objects (in this case, FieldPermissions). You can do this from the PermissionSet object by using the child relationship (`ObjectPerms`, `FieldPerms`, and so on) or from the child object by referencing the PermissionSet with `Parent.permission_set_attribute`.

It's important to consider when to use a conditional `WHERE` statement to restrict the result set. To query based on an attribute on the permission set object, nest the SOQL with the child relationship. However, to query based on an attribute on the child object, you must reference the permission set parent attribute in your query.

The following two queries return the same columns with different results, based on whether you use the child relationship or parent notation.

```
SELECT PermissionsEditReadOnlyFields,
       (SELECT SObjectType, Field, PermissionsRead, PermissionsEdit
        FROM FieldPerms
        WHERE SObjectType = 'Merchandise__c')
       FROM PermissionSet
      WHERE PermissionsEditReadOnlyFields = true
```

versus:

```
SELECT SObjectType, Field, PermissionsRead, PermissionsEdit, Parent.Name,
       Parent.PermissionsEditReadOnlyFields
      FROM FieldPermissions
     WHERE SObjectType='Merchandise__c'
```

SEE ALSO:

[PermissionSet](#)

[ObjectPermissions](#)

FiscalYearSettings

Settings to define a custom or standard fiscal year for your organization. This object has a parent-child relationship with the Period object.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Description of the setting.</p>

Field	Details
EndDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description End date of the fiscal year.</p>
IsStandardYear	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the fiscal year is a standard calendar year (<code>true</code>) or a custom fiscal year (<code>false</code>).</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description A name for the fiscal year. Limit: 80 characters.</p>
PeriodId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the associated fiscal period.</p>
PeriodLabelScheme	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The numbering scheme used for fiscal periods.</p>
PeriodPrefix	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Sort</p>

Field	Details
QuarterLabelScheme	Description The prefix of fiscal periods. For example, if "P" is the prefix, then the first period is "P1."
QuarterPrefix	Type picklist
Properties	Filter, Group, Nillable, Restricted picklist, Sort
Description	The numbering scheme used for fiscal quarters.
StartDate	Type date
Properties	Filter, Group, Nillable, Sort
Description	The prefix of fiscal quarters. For example, if "Q" is the prefix, then the fourth quarter would be "Q4."
WeekLabelScheme	Type picklist
Properties	Filter, Group, Nillable, Restricted picklist, Sort
Description	Start date of the fiscal year.
WeekStartDay	Type int
Properties	Filter, Group, Nillable, Sort
Description	The name of the day that starts the week, for example Monday or Sunday
YearType	Type picklist

Field	Details
	Properties Filter, Group, Nillable, Restricted picklist, Sort
	Description Indicates one of two types of fiscal years, Standard or Custom. Standard denotes the standard Gregorian calendar, while Custom means a fiscal year with a custom structure.

SEE ALSO:

[Period](#)
[Object Basics](#)

FlexQueueItem

Represents an asynchronous Apex job in the Apex flex queue. Provides information about the job type and flex queue position of the AsyncApexJob. This object is available in API version 36.0 and later.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field Name	Description
AsyncApexJobId	Type reference Properties Filter, Group, Sort Description The ID of an AsyncApexJob that's waiting in the flex queue.
FlexQueueItemId	Type string Properties Filter, Group, Nillable, Sort Description The primary key for this FlexQueueItem.
JobPosition	Type int

Field Name	Description
	<p>Properties Filter, Group, Sort</p> <p>Description The position in the flex queue of the waiting job. The highest-priority job in the queue is at position 0.</p>
JobType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of the job. Valid values are:</p> <ul style="list-style-type: none"> • ApexToken • BatchApex • BatchApexWorker • Future • Queueable • ScheduledApex • SharingRecalculation • TestRequest • TestWorker <p>Currently, queries are supported only on <code>BatchApex</code> jobs.</p>

Usage

To find the position of an `AsyncApexJob` in the flex queue, query `JobPosition`. For example:

```
SELECT JobPosition FROM FlexQueueItem WHERE JobType = 'BatchApex' AND AsyncApexJobId = '707xx000000DABC'
```

To find the job at a given position, query `AsyncApexJobId`. For example:

```
SELECT AsyncApexJobId FROM FlexQueueItem WHERE JobType = 'BatchApex' AND JobPosition = '2'
```

To find all batch jobs in the flex queue, query `JobType`. To get other information about the jobs, include `AsyncApexJob` in your query. For example:

```
SELECT JobType, JobPosition, AsyncApexJob.ApexClass.Name, AsyncApexJob.CreatedDate, AsyncApexJob.CreatedBy FROM FlexQueueItem WHERE JobType='BatchApex' AND AsyncApexJob.ApexClass.Name LIKE '%BatchAJob%' ORDER BY JobPosition DESC
```

FlowInterview

Represents a flow interview. A *flow interview* is a running instance of a flow.

Supported Calls

`delete(), describeLayout(), getDeleted(), getUpdated(), query(), retrieve(), describeSObject()`

Special Access Rules

To delete a flow interview, you must have the “Manage Force.com Flow” user permission. All other calls require the “Run Flows” user permission or the `Force.com Flow User` field enabled on the user detail page.

Fields

Field Name	Details
CurrentElement	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The flow element at which the interview is currently paused or waiting.</p>
InterviewLabel	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Label for the interview. This label helps users and administrators differentiate interviews from the same flow. In the user interface, this label appears in the Paused Flow Interviews component on the user’s Home tab and in the Paused and Waiting Interviews list on the flow management page.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name for the interview.</p>
OwnerId	<p>Type ID</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The ID of the user who owns the interview. Only this user or an admin can resume the interview.</p>
PauseLabel	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Information about why the interview was paused. This string is entered by the user who paused the flow interview. Label is Why Paused</p>

FlowInterviewOwnerSharingRule

Represents the rules for sharing a FlowInterview with users other than the owner. This object is available in API version 33.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description A description of the sharing rule. Maximum size is 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group that's given access.</p>

Usage

Use this object to manage the sharing rules for FlowInterview records. Sharing rules don't affect who can resume the FlowInterview. Only the owner or a flow admin can resume a FlowInterview. General sharing uses this object.

FlowInterviewShare

Represents a sharing entry on a FlowInterview. This object is available in API version 33.0 and later.

Supported Calls

`describeSObjects(), create(), delete(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the FlowInterview. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All This value is not valid for creating or deleting records.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the FlowInterview associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to Manual (default).</p>

Field	Details
	<p>Valid values include:</p> <ul style="list-style-type: none">• Manual—The User or Group has access because a user with “All” access manually shared the FlowInterview with them.• Owner—The User is the owner of the FlowInterview.• Rule—The User or Group has access via a FlowInterview sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the FlowInterview. This field can't be updated.</p>

Usage

This object lets you determine which users and groups can view and edit FlowInterview records owned by other users. However, it doesn't let you determine who can resume the FlowInterview. Only the owner or a flow admin can resume a FlowInterview.

Folder

Represents a repository for a Document, EmailTemplate, Report, or Dashboard. Only one type of item can be contained in a folder.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- You must have the “Modify All Data” permission to create, update, or delete document folders and email template folders.
- Customer Portal users can't access this object.
- To query this object, no special permissions are needed.
- As of API version 35.0, when a folder is shared with a role, it is only visible to users in that role. Superior roles in the role hierarchy don't gain visibility.
- If analytics folder sharing is turned on, then users need these permissions to create and manage report folders and dashboard folders:
 - “Create Dashboard Folders”
 - “Create Report Folders”

Fields

Field	Details
AccessType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. Indicates who can access the Folder. Available values include:</p> <ul style="list-style-type: none"> • Hidden—Folder is hidden from everyone. • Public—Folder is accessible by all users. • Shared—Folder is accessible only by a User in a particular Group or UserRole. The API doesn't allow you to view, insert, or update which group or Role the Folder is shared with. <p> Note: If analytics folder sharing is turned on for your organization, then this field is present but not used.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Folder Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Isreadonly	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this Folder is read-only (<code>true</code>) or editable (<code>false</code>). Label is <i>Read Only</i>.</p> <p> Note: If analytics folder sharing is turned on for your organization, then this field is present but not used.</p>

Field	Details
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the folder as it appears in the user interface. Label is Document Folder Label.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Required. Type of objects contained in the Folder. This field can't be updated. Available values include:</p> <ul style="list-style-type: none"> Dashboard Document Email template Report

Usage

Only one type of item can be contained in a folder, either Document, EmailTemplate, Report, or Dashboard.

SEE ALSO:

[Object Basics](#)

FolderedContentDocument

Represents the relationship between a parent and child ContentFolderItem in a ContentWorkspace.

Supported Calls

`describeSObjects()`

Fields

Field Name	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the ContentDocument that can be in a folder.</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description File size of the ContentDocument.</p>
FileExtension	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description File extension of the ContentDocument.</p>
FileType	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description File type of the ContentDocument.</p>
IsFolder	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates that the FolderedContentDocument is a folder, rather than a file.</p>
ParentContentFolderId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the ContentFolder the ContentDocument resides in.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Name of the file or folder in a ContentFolder.</p>

ForecastingAdjustment

This object represents an individual sales manager's adjustment for a subordinate's forecast via a ForecastingItem. Available in API versions 26 and greater. This object is separate from the ForecastingOwnerAdjustment object, which represents forecast users' adjustments of their own forecasts.

 **Note:** This information only applies to Collaborative Forecasts.

Supported Calls

```
create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), update(), upsert()
```

Fields

Field Name	Details
AdjustedAmount	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The revenue amount of an individual forecast item, after an adjustment.</p>
AdjustedQuantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The quantity amount of an individual forecast item, after an adjustment. This field is available in API version 28 and later.</p>
AdjustmentNote	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A text note providing information about the adjustment. The maximum length is 140 characters. This field does not appear in reports.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The currency code of the adjustment. If omitted, the default is the importing user's personal currency.</p>
ForecastCategoryName	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The category within the sales cycle that an opportunity is assigned to based on its opportunity stage. The standard forecast categories are Pipeline, Best Case,</p>

Field Name	Details
ForecastingItemCategory	<p>Commit, Omitted, and Closed. You can customize forecast category names. The forecast categories display information for that specific category; for example, Best Case only reflects amounts in the Best Case category.</p>
	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description This field indicates which type of forecast rollup the manager adjustment belongs to. Depending on whether your organization uses individual forecast category rollups or cumulative forecast rollups, you have these possible values for the ForecastingItemCategory field.</p> <p>Individual forecast category rollups:</p> <ul style="list-style-type: none"> • PipelineOnly - Rollup from Pipeline opportunities only. • BestCaseOnly - Rollup from Best Case opportunities only. Adjustable. • CommitOnly - Rollup from Commit opportunities only. Adjustable. <p>Cumulative forecast rollups:</p> <ul style="list-style-type: none"> • OpenPipeline - Rollup from Pipeline + Best Case + Commit opportunities. • BestCaseForecast - Rollup from Best Case + Commit + Closed opportunities. Adjustable. • CommitForecast - Rollup from Commit + Closed opportunities. Adjustable. <p>Either cumulative or individual forecast category rollups:</p> <ul style="list-style-type: none"> • ClosedOnly - Rollup from Closed opportunities only. <p>The ForecastingItemCategory field differs from the ForecastCategoryName field.</p> <ul style="list-style-type: none"> • The ForecastCategoryName field represents the forecast category of the <i>underlying opportunities</i> rolling up to forecast amounts. In organizations using cumulative forecast rollups, the ForecastCategoryName field can be null because the cumulative forecast amounts include opportunities from multiple forecast categories. • The new ForecastingItemCategory field represents the <i>type of rollup</i> a forecast amount or adjustment is from. In organizations using individual forecast category columns, it contains the individual forecast rollup categories. In organizations using cumulative forecast rollups, it contains the cumulative rollup categories. <p>When inserting manager adjustments, the values you insert for ForecastCategoryName and ForecastingItemCategory must be compatible with each other. In organizations using cumulative forecast rollups, the ForecastCategoryName is nullable. These are the valid pairs.</p>

Field Name	Details
Individual forecast category rollups:	
	<ul style="list-style-type: none"> • ForecastCategoryName: BestCase, ForecastingItemCategory: BestCaseOnly • ForecastCategoryName: Commit, ForecastingItemCategory: CommitOnly
Cumulative forecast category rollups:	
	<ul style="list-style-type: none"> • ForecastCategoryName: null, ForecastingItemCategory: BestCaseForecast • ForecastCategoryName: null, ForecastingItemCategory: CommitForecast
ForecastingTypeId	<p>Type reference</p>
Properties	
Create, Filter, Group, Sort	
Description	
The ID of the related ForecastingType.	
ForecastingItemId	<p>Type reference</p>
Properties	
Filter, Group, Sort	
Description	
The ID of the related ForecastingItem.	
IsAmount	<p>Type boolean</p>
Properties	
Defaulted on create, Filter, Group, Sort	
Description	
True indicates that the adjustment is made in a revenue amount. If false, then IsQuantity must be true. This field is available in API version 28 and later.	
IsQuantity	<p>Type boolean</p>
Properties	
Defaulted on create, Filter, Group, Sort	
Description	
True indicates that the adjustment is made in a quantity amount. If false, then IsAmount must be true. This field is available in API version 28 and later.	

Field Name	Details
OwnerId	<p>Type reference</p> <p>Properties Create,Defaulted on create, Filter, Group, Sort</p> <p>Description The ID of the forecast owner.</p>
PeriodId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Period ID for the adjustment. Read only.</p>
ProductFamily	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The Product Family for the adjustment. Read only. This field is available in API version 29 and later.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The start of the adjustment, expressed as month and year. The date can include any day in a given month. Stored using the first date of the month.</p>

Usage

Use this object to obtain a manager's adjustment detail for a specified ForecastingItem. The ForecastingAdjustment object itself is visible to all users, but only forecast managers and users above them in the forecast hierarchy can read or write ForecastingAdjustment records.

 **Note:** Beginning with API version 30.0, organizations can have more than one forecasting type enabled. The ForecastingQuota, ForecastingAdjustment, ForecastingOwnerAdjustment, ForecastingItem, and ForecastingFact objects can all have records with different ForecastingTypeId values. Use the ForecastingType

object to determine the ID for each forecast type and then filter `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingItem`, or `ForecastingFact` records as necessary.

SEE ALSO:

[ForecastingFact](#)
[ForecastingItem](#)
[ForecastingQuota](#)

ForecastingFact

This is a read-only object linking a `ForecastingItem` with its opportunities, such as opportunities that share the same owner or forecast category and have a closing date within the period of the forecasting item. Available in API versions 26 and greater.

 **Note:** This information only applies to Collaborative Forecasts.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
<code>ForecastCategoryName</code>	Type picklist Properties Filter, Group, Restricted picklist, Sort Description A forecast category is the category within the sales cycle to which an opportunity is assigned based on its opportunity stage. The standard forecast categories are Pipeline, Best Case, Commit, Omitted, and Closed. You can customize the forecast category names.
<code>ForecastedObjectId</code>	Type reference Properties Filter, Group, Nillable, Sort Description Contains the Split ID of the forecasted OpportunitySplit object if the forecast data source is opportunity splits or the OpportunityLineItem ID of the forecasted opportunity if the data source is product families. If the data source is product families and the opportunity has no line item, this field is Null. If the forecast data source is opportunities, this field is Null. This field is available in API version 29 and later. Read only.

Field Name	Details
ForecastingItemId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the ForecastingItem.</p>
ForecastingTypeId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the related ForecastingType.</p>
OpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The opportunity ID.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the opportunity owner.</p>
PeriodId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Period ID for the forecast.</p>

Usage

Use this object to get information about opportunities linked to forecasting items.



Note: Beginning with API version 30.0, organizations can have more than one forecasting type enabled. The `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingOwnerAdjustment`, `ForecastingItem`, and `ForecastingFact` objects can all have records with different `ForecastingTypeId` values. Use the `ForecastingType` object to determine the ID for each forecast type and then filter `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingItem`, or `ForecastingFact` records as necessary.

S

SEE ALSO:

[ForecastingAdjustment](#)

[ForecastingItem](#)

[ForecastingQuota](#)

ForecastingItem

This is a read-only object used for individual forecast amounts. Users see amounts based on their perspectives and forecast roles. The amounts users see include one of the following when forecasting in revenue: `AmountWithoutAdjustments`, `AmountWithoutManagerAdjustment`, `ForecastAmount`, `OwnerOnlyAmount`. The amounts users see include one of the following when forecasting in quantity: `QuantityWithoutAdjustments`, `QuantityWithoutManagerAdjustment`, `ForecastQuantity`, `OwnerOnlyQuantity`. Available in API versions 26 and greater.

Additionally, note that users:

- with the “View All Forecasts” permission have access to all `ForecastingItem` fields.
- without the “View All Forecasts” permission have access to all fields for their own subordinates.

Other users can see the `ForecastingItem` object, but not its records.



Note: This information only applies to Collaborative Forecasts.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
<code>AmountWithoutAdjustments</code>	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The sum of a person’s owned revenue opportunities and the person’s subordinates’ opportunities, without adjustments. Subordinates include everyone reporting up to a person in the forecast hierarchy. This amount is visible only on reports.</p>

Field Name	Details
AmountWithoutManagerAdjustment	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The forecast number as seen by the forecast owner. This is the sum of the owner's revenue opportunities and the owner's subordinates' opportunities, including adjustments made by the forecast owner on the owner's or subordinates' forecasts. It doesn't include adjustments made by forecast managers above the owner in the forecast hierarchy.</p>
AmountWithoutOwnerAdjustment	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The forecast amount as seen by the forecast owner without the owner's adjustment. This is the sum of the subordinate's opportunities, including adjustments made by their manager or by the subordinate themselves, plus the rollup of the owner's own opportunities. <i>It doesn't include adjustments made by the forecast owner.</i></p>
ForecastAmount	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The revenue forecast from the forecast manager's perspective and the sum of the owner's and subordinates' opportunities, including all forecast adjustments.</p>
ForecastCategoryName	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description A forecast category is the category within the sales cycle to which an opportunity is assigned based on its opportunity stage. The standard forecast categories are Pipeline, Best Case, Commit, Omitted, and Closed. You can customize the forecast category names.</p>
ForecastQuantity	<p>Type double</p>

Field Name	Details
	Properties Filter, Sort, Nillable
ForecastingItemCategory	Description <p>The quantity forecast from the forecast manager's perspective and the sum of the owner's and subordinates' opportunities, including all forecast adjustments. This field is available in API version 28 and later.</p>
ForecastingItemCategory	Type picklist
	Properties Filter, Group, Sort
	Description <p>This field indicates which type of forecast rollup the forecasting item belongs to. Depending on whether your organization uses individual forecast category rollups or cumulative forecast rollups, you have these possible values for the ForecastingItemCategory field.</p>
	Individual forecast category rollups:
	<ul style="list-style-type: none"> • PipelineOnly - Rollup from Pipeline opportunities only. • BestCaseOnly - Rollup from Best Case opportunities only. Adjustable. • CommitOnly - Rollup from Commit opportunities only. Adjustable.
	Cumulative forecast rollups:
	<ul style="list-style-type: none"> • OpenPipeline - Rollup from Pipeline + Best Case + Commit opportunities. • BestCaseForecast - Rollup from Best Case + Commit + Closed opportunities. Adjustable. • CommitForecast - Rollup from Commit + Closed opportunities. Adjustable.
	Either cumulative or individual forecast category rollups:
	<ul style="list-style-type: none"> • ClosedOnly - Rollup from Closed opportunities only.
	<p>The ForecastingItemCategory field differs from the ForecastCategoryName field.</p>
	<ul style="list-style-type: none"> • The ForecastCategoryName field represents the forecast category of the <i>underlying opportunities</i> rolling up to forecast amounts. In organizations using cumulative forecast rollups, the ForecastCategoryName field can be null because the cumulative forecast amounts include opportunities from multiple forecast categories. • The new ForecastingItemCategory field represents the <i>type of rollup</i> a forecast amount or adjustment is from. In organizations using individual forecast category columns, it contains the individual forecast rollup categories. In organizations using cumulative forecast rollups, it contains the cumulative rollup categories.

Field Name	Details
ForecastingTypeID	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the related ForecastingType.</p>
HasAdjustment	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description A flag that indicates if the forecasting item includes a <i>manager</i> adjustment. This flag is true only when the item includes an adjustment and the user performing the query has read access to the adjustment.</p>
HasOwnerAdjustment	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description A flag that indicates if the forecasting item includes an <i>owner</i> adjustment. This flag is true only when the item includes an adjustment and the user performing the query has read access to the adjustment. Available in API versions 33 and greater.</p>
IsAmount	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description True indicates that the adjustment is made in a revenue amount. If false, then <code>IsQuantity</code> must be true. This field is available in API version 28 and later.</p>
IsQuantity	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description True indicates that the adjustment is made in a quantity amount. If false, then <code>IsAmount</code> must be true. This field is available in API version 28 and later.</p>

Field Name	Details
IsUpToDate	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description A flag indicating whether or not a specific forecasting item reflects current information. For example, if users are making adjustments which are in process, the item won't be up-to-date.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the forecast owner.</p>
OwnerOnlyAmount	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The sum of a person's revenue opportunities, without adjustments.</p>
OwnerOnlyQuantity	<p>Type double</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The sum of a person's quantity opportunities, without adjustments. This field is available in API version 28 and later.</p>
ParentForecastingItemId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the ForecastingItem that the current item rolls up to.</p>
PeriodId	<p>Type reference</p>

Field Name	Details
	Properties
	Filter, Group, Nillable, Sort
	Description
	Period ID for the forecast.
ProductFamily	Type
	picklist
	Properties
	Filter, Group, Nillable, Sort
	Description
	The product family of the forecast item. This field is available in API version 29 and later. Read only.
QuantityWithoutAdjustments	Type
	double
	Properties
	Filter, Sort, Nillable
	Description
	The sum of a person's owned quantity opportunities and also his or her subordinates' opportunities, without adjustments. Subordinates include everyone reporting up to a person in the forecast hierarchy. This field is available in API version 28 and later.
QuantityWithoutManagerAdjustment	Type
	double
	Properties
	Filter, Sort, Nillable
	Description
	The forecast number as seen by the forecast owner. This is the sum of the owner's quantity opportunities and subordinates' opportunities, including adjustments made on the subordinates' forecasts. It doesn't include adjustments made by forecast managers above the owner in the forecast hierarchy. This field is available in API version 28 and later.
QuantityWithoutOwnerAdjustment	Type
	double
	Properties
	Filter, Sort, Nillable
	Description
	The forecast quantity as seen by the forecast owner without the owner's adjustment. This is the sum of the subordinate's opportunities, including

Field Name	Details
	adjustments made by their manager or by the subordinate themselves, plus the rollup of the owner's own opportunities. <i>It doesn't include adjustments made by the forecast owner.</i> This field is available in API version 38.0 and later.

Usage

Use this object to obtain individual forecast amounts, either with or without adjustments, based on a user's perspective and forecast role. The ForecastingItem object is visible to all users, but only forecast managers and users above them in the forecast hierarchy can read or write ForecastingAdjustment records.

 **Note:** Beginning with API version 30.0, organizations can have more than one forecasting type enabled. The ForecastingQuota, ForecastingAdjustment, ForecastingOwnerAdjustment, ForecastingItem, and ForecastingFact objects can all have records with different ForecastingTypeId values. Use the ForecastingType object to determine the ID for each forecast type and then filter ForecastingQuota, ForecastingAdjustment, ForecastingItem, or ForecastingFact records as necessary.

SEE ALSO:

[ForecastingAdjustment](#)
[ForecastingFact](#)
[ForecastingQuota](#)

ForecastingOwnerAdjustment

This object represents an individual forecast user's adjustment of their *own* forecast via a ForecastingItem. Available in API versions 33 and greater. This object is separate from the ForecastingAdjustment object, which represents managers' adjustments of *subordinates'* forecasts.

 **Note:** This information only applies to Collaborative Forecasts.

Supported Calls

`create(), delete(), describesObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>

Field Name	Details
	Description
	<p>The currency code of the adjustment. If omitted, the default is the importing user's personal currency.</p>
ForecastCategoryName	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort
	Description
	<p>The category within the sales cycle that an opportunity is assigned to based on its opportunity stage. The standard forecast categories are Pipeline, Best Case, Commit, Omitted, and Closed. You can customize forecast category names. The forecast categories display information for that specific category; for example, Best Case only reflects amounts in the Best Case category.</p>
ForecastingItemCategory	Type
	picklist
	Properties
	Create, Filter, Group, Sort
	Description
	<p>This field indicates which type of forecast rollup the owner adjustment belongs to. Depending on whether your organization uses individual forecast category rollups or cumulative forecast rollups, you have these possible values for the ForecastingItemCategory field.</p>
	<i>Individual forecast category rollups:</i>
	<ul style="list-style-type: none"> • PipelineOnly - Rollup from Pipeline opportunities only. • BestCaseOnly - Rollup from Best Case opportunities only. Adjustable. • CommitOnly - Rollup from Commit opportunities only. Adjustable.
	<i>Cumulative forecast rollups:</i>
	<ul style="list-style-type: none"> • OpenPipeline - Rollup from Pipeline + Best Case + Commit opportunities. • BestCaseForecast - Rollup from Best Case + Commit + Closed opportunities. Adjustable. • CommitForecast - Rollup from Commit + Closed opportunities. Adjustable.
	<i>Either cumulative or individual forecast category rollups:</i>
	<ul style="list-style-type: none"> • ClosedOnly - Rollup from Closed opportunities only.
	<p>The ForecastingItemCategory field differs from the ForecastCategoryName field.</p>
	<ul style="list-style-type: none"> • The ForecastCategoryName field represents the forecast category of the <i>underlying opportunities</i> rolling up to forecast amounts. In organizations using cumulative forecast rollups, the ForecastCategoryName field

Field Name	Details
ForecastingItemId	<p>can be null because the cumulative forecast amounts include opportunities from multiple forecast categories.</p> <ul style="list-style-type: none"> The new <code>ForecastingItemCategory</code> field represents the <i>type of rollup</i> a forecast amount or adjustment is from. In organizations using individual forecast category columns, it contains the individual forecast rollup categories. In organizations using cumulative forecast rollups, it contains the cumulative rollup categories. <p>When inserting owner adjustments, the values you insert for <code>ForecastCategoryName</code> and <code>ForecastingItemCategory</code> must be compatible with each other. In organizations using cumulative forecast rollups, the <code>ForecastCategoryName</code> is nullable. These are the valid pairs.</p> <p>Individual forecast category rollups:</p> <ul style="list-style-type: none"> <code>ForecastCategoryName: BestCase,</code> <code>ForecastingItemCategory: BestCaseOnly</code> <code>ForecastCategoryName: Commit,</code> <code>ForecastingItemCategory: CommitOnly</code> <p>Cumulative forecast category rollups:</p> <ul style="list-style-type: none"> <code>ForecastCategoryName: null, ForecastingItemCategory: BestCaseForecast</code> <code>ForecastCategoryName: null, ForecastingItemCategory: CommitForecast</code>
ForecastingTypeId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the related ForecastingItem.</p>
ForecastOwnerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the related ForecastingType.</p>

Field Name	Details
	<p>Description</p> <p>The ID of the forecast owner.</p>
IsAmount	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>True indicates that the adjustment is made in a revenue amount. If false, then <code>IsQuantity</code> must be true.</p>
IsQuantity	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>True indicates that the adjustment is made in a quantity amount. If false, then <code>IsAmount</code> must be true.</p>
OwnerAdjustedAmount	<p>Type</p> <p>currency</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The revenue amount of an individual forecast item, after an adjustment.</p>
OwnerAdjustedQuantity	<p>Type</p> <p>double</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The quantity amount of an individual forecast item, after an adjustment.</p>
OwnerAdjustmentNote	<p>Type</p> <p>textarea</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>A text note providing information about the adjustment. The maximum length is 140 characters. This field does not appear in reports.</p>

Field Name	Details
PeriodId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Period ID for the adjustment. Read only.</p>
ProductFamily	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The Product Family for the adjustment. Read only.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The start of the adjustment, expressed as month and year. The date can include any day in a given month. Stored using the first date of the month.</p>

Usage

Use this object to obtain a user's adjustment detail for a specified ForecastingItem in their own forecast.



Note: Beginning with API version 30.0, organizations can have more than one forecasting type enabled. The `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingOwnerAdjustment`, `ForecastingItem`, and `ForecastingFact` objects can all have records with different `ForecastingTypeId` values. Use the `ForecastingType` object to determine the ID for each forecast type and then filter `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingItem`, or `ForecastingFact` records as necessary.

ForecastingQuota

This object represents an individual user's quota for a specified time period. The "Manage Quotas" user permission is required for creating, updating, or deleting quotas. (Users can only edit their subordinates' quotas, not their own.) The "View All Forecasts" permission is required to View any user's forecast, regardless of the forecast role hierarchy. Available in API versions 25 and greater. Forecast managers can view the forecasts of subordinates who report to them in the forecast hierarchy.



Note: This information only applies to Collaborative Forecasts.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The currency code of the quota. If omitted, the default is the importing user's personal currency.</p>
ForecastingTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the related ForecastingType.</p>
IsAmount	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description True indicates that the adjustment is made in a revenue amount. If false, then <code>IsQuantity</code> must be true. This field is available in API version 28 and later.</p>
IsQuantity	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description True indicates that the adjustment is made in a quantity amount. If false, then <code>IsAmount</code> must be true. This field is available in API version 28 and later.</p>
PeriodId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Period ID for the quota. Read only.</p>
ProductFamily	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The product family for the quota. This field is available in API version 29 and later.</p>
QuotaAmount	<p>Type currency</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The revenue quota amount for an individual user and for a specific period.</p>
QuotaOwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID that identifies the quota owner.</p>
QuotaQuantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The quantity quota amount for an individual user and for a specific period. This field is available in API version 28 and later.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	The start of the quota, expressed as month and year. The date can include any day in a given month. Stored using the first date of the month.

Usage

Use this object to get an individual user's quota for a specified time period.

 **Note:** Beginning with API version 30.0, organizations can have more than one forecasting type enabled. The `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingOwnerAdjustment`, `ForecastingItem`, and `ForecastingFact` objects can all have records with different `ForecastingTypeId` values. Use the `ForecastingType` object to determine the ID for each forecast type and then filter `ForecastingQuota`, `ForecastingAdjustment`, `ForecastingItem`, or `ForecastingFact` records as necessary.

SEE ALSO:

[ForecastingAdjustment](#)

[ForecastingFact](#)

[ForecastingItem](#)

ForecastingType

This object is used to identify the forecast type associated with `ForecastingAdjustment`, `ForecastingOwnerAdjustment`, `ForecastingQuota`, `ForecastingFact`, and `ForecastingItem` objects. Available in API versions 30.0 and greater.

 **Note:** This information only applies to Collaborative Forecasts.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
DeveloperName	Type string Properties Filter, Group, Sort
	Description
	<p>The name of the forecasting type. Allowed values include:</p> <ul style="list-style-type: none"> • <code>OpportunityRevenue</code> : Opportunities - Revenue

Field Name	Details
	<ul style="list-style-type: none"> • OpportunityQuantity : Opportunities - Quantity • OpportunitySplitRevenue : Opportunity Revenue Splits - Revenue • OpportunityOverlayRevenue : Opportunity Overlay Splits - Revenue • OpportunityLineItemRevenue : Product Families - Revenue • OpportunityLineItemQuantity : Product Families - Quantity • The name of a custom opportunity split type that has been enabled as a forecast type. Custom split types are based on currency fields, which can contain revenue amounts only. <p>The <code>DeveloperName</code> is called <code>name</code> in the Metadata API and Forecasting Type in custom reports.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the forecasting type is currently enabled in the organization.</p>
IsAmount	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the forecasting type is based on the revenue measurement.</p>
IsQuantity	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the forecasting type is based on the quantity measurement.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of the forecasting type.</p>

Field Name	Details
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Master label for this Forecasting Type value. This display value is the internal label that does not get translated.</p>

Usage

Use this object to identify the forecast type of `ForecastingAdjustment`, `ForecastingQuota`, `ForecastingFact`, and `ForecastingItem` objects.

ForecastShare

Represents the sharing of a customizable forecast at a given role and territory.

 **Note:** This information applies to Customizable Forecasting and not [Collaborative Forecasts](#).

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description A value (Read, Edit, or All) that represents the type of sharing being allowed.</p>

Field	Details
CanSubmit	<p>Type boolean</p> <p>Properties Filter, Group, Sort</p> <p>Description Indicates whether the user or group can submit forecasts (<code>True</code>) or not (<code>False</code>).</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none"> • <code>Manual</code>—The User or Group has access because a User has manually shared the forecast with them. • <code>Owner</code>—The User is the owner of the forecast.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID representing the User or Group being granted access.</p>
UserRoleId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the UserRole associated with this object.</p>

Usage

This object allows you to determine which users and groups can view or submit forecasts owned by other users.

Goal

The Goal object represents the components of a goal such as its name, description, and status.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
CompletionDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The completion date of the goal.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the goal. The maximum length is 65,535 characters.</p>
DueDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The date the goal is due.</p>
ImageUrl	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The URL for the goal image. The image must be stored in Documents and set as externally available. Applicable only to Goal objects of Type: Goal.</p>
IsKeyCompanyGoal	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the goal is a key company goal.</p>

Field Name	Details
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp that indicates when a user last viewed a record that is related to this goal.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp that indicates when a user last viewed this goal. If this value is null, this record might have been only referenced (LastReferencedDate) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the goal. The maximum length is 255 characters.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns the goal.</p>
Progress	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The progress of the goal measured as a percentage.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The start date of the goal.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The status of the goal.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Draft • Published • Completed • Canceled • Not Completed

GoalFeed

Represents a single feed item in the feed displayed for a Goal record. The goal feed shows changes to a goal for fields that are tracked in feeds, posts, and comments about the goal.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The numbers of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting ContentFileName automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Available in API version 36.0 and earlier only. Group, Nillable, Sort</p> <p>Description Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p>

Field Name	Details
	Properties
	Available in API version 36.0 and earlier only. Group, Nillable, Sort
	Description
	Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then <code>InsertedById</code> is set to the ID of the logged-in user.
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.
	Rich text supports the following HTML tags:
	<ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field Name	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the goal record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API.</p> <ul style="list-style-type: none">ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.AdvancedTextPost—created when a user posts a group announcement.AnnouncementPost—Not used.ApprovalPost—generated when a user submits an approval.BasicTemplateFeedItem—Not used.CanvasPost—a post made by a canvas app posted on a feed.CollaborationGroupCreated—generated when a user creates a public group.CollaborationGroupUnarchived—Not used.ContentPost—a post with an attached file.CreatedRecordEvent—generated when a user creates a record from the publisher.DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.LinkPost—a post with an attached URL.PollPost—a poll posted on a feed.ProfileSkillPost—generated when a skill is added to a user's Chatter profile.QuestionPost—generated when a user posts a question.ReplyPost—generated when Chatter Answers posts a reply.

Field Name	Details
	<ul style="list-style-type: none"> • <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. • <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case.

GoalHistory

This read-only object contains historical information about changes that have been made to the Goal object.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
<code>Field</code>	Type picklist

Field Name	Details
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	The name of the field that was changed.
GoalId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	ID of the goal.
NewValue	Type
	anyType
	Properties
	Nillable, Sort
	Description
	The new value of the field that was changed.
OldValue	Type
	anyType
	Properties
	Nillable, Sort
	Description
	The latest value of the field before it was changed.

GoalLink

Represents the relationship between two goals. This is a many-to-many relationship, meaning that each goal can link to many other goals.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The auto-generated name of the goal link.</p>
ParentGoalId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the parent goal.</p>
SubgoalId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the subgoal.</p>

GoalShare

Represents a sharing entry on a Goal object.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the goal. The possible values are:</p> <ul style="list-style-type: none">• Read Only• Read/Write• Owner <p>This field must be set to an access level that is higher than the organization's default access level for goals.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the goal that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none">• Owner: User is the owner of the Goal or is in a user role above the Goal owner in the role hierarchy.• Manual: User or group has access because a user with "All" access manually shared the Goal with the user or group.• Rule: User or group has access via a Goal sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the Goal. This field can't be updated.</p>

GoogleDoc

Represents a link to a Google Document. This object is available in API version 14.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()`

Special Access Rules

This object is available in **All** Editions except **Database.com** for Google Apps Premier Edition accounts. See the Salesforce online help for more information.

Fields

Field	Details
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the Google document.</p>
Owner	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Update</p> <p>Description The ID of the user who currently owns this Google Document. Default value is the user logged in to the API to perform the create.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the attachment's parent object. The following objects are supported as parents of Google documents: Account, Asset, Campaign, Case, Contact, Contract, Custom Objects, Lead, Opportunity, Product2, and Solution.</p>
Url	<p>Type string</p>

Field	Details
	Properties Create, Filter, Nullable, Update
	Description The URL of the Google document.

Group

A set of User records.

Groups are sets of users. They can contain individual users, other groups, the users in a particular role or territory, or the users in a particular role or territory plus all of the users below that role or territory in the hierarchy.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), search(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
DefaultDivision	Type picklist Properties Filter, Group, Restricted picklist, Sort Description This record's default division. Only applicable if divisions are enabled for your organization.
DeveloperName	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed

Field	Details
	<p>package and the changes are reflected in a subscriber's organization. Corresponds to Group Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
DoesIncludeBosses	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the managers have access (<code>true</code>) or do not have access (<code>false</code>) to records shared with members of the group. This field is only available for public groups. This field is available in API version 18.0 and later.</p>
DoesSendEmailToMembers	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the email is sent (<code>true</code>) or not sent (<code>false</code>) to the group members. The email is sent to queue members as well.</p>
Email	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Email address for a group of type Case. Applies only for a case queue.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of the group. Corresponds to Label on the user interface.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description ID of the user who owns the group.</p>
QueueRoutingConfig	<p>Type reference</p> <p>Properties Create, Delete, Query, Retrieve, Update</p> <p>Description The ID of the queue routing configuration associated with the queue.</p>
RelatedId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the ID of the associated groups. For Groups of type “Role,” the ID of the associated UserRole. The RelatedId field is polymorphic.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Required. Type of the Group. One of the following values:</p> <ul style="list-style-type: none">• Regular—Standard Public Group. When you <code>create()</code> a Group, its type must be <code>Regular</code>, unless a partner portal is enabled for the organization, in which case the type can be <code>Regular</code> or <code>PRMOrganization</code>.• Role—Public Group that includes all of the User records in a particular UserRole.• RoleAndSubordinates—Public Group that includes all of the User records in a particular UserRole and all of the User records in any subordinateUserRole.• Organization—Public Group that includes all of the User records in the organization. This Group is read-only.• Case—Public group of users, members of a queue that can own a Case.• Lead—Public group of users, members of a queue that can own a Lead.• Manager—Public group that includes a user’s direct and indirect managers. This Group is read-only.• ManagerAndSubordinatesInternal—Public Group that includes a user and the user’s direct and indirect reports. This Group is read-only.• PRMOrganization—Public Group that includes all of the partners in an organization that has the partner portal feature enabled.

Field	Details
	<ul style="list-style-type: none">• Queue—Public Group that includes all of the User records that are members of a queue.• Territory—Public Group that includes all of the User records in an organization that has the territory feature enabled.• TerritoryAndSubordinates—Public Group that includes all of the User records in a particular UserRole and all of the User records in any subordinateUserRole.• Collaboration—Chatter group. <p>Only <code>Regular</code>, <code>Case</code>, and <code>Lead</code> can be used when creating a group. The other values are reserved.</p>

Usage

Unlike users, this object can be deleted. Any User can access this object—no special permissions are needed.

Only public groups are accessible via the API. Personal groups are not available.

In API version 38.0 and later, you can query a Group using `Related.Name` to retrieve the Group's name. `Related.Name` is supported for public groups, user roles, territories, manager groups, and user names.

In API version 13.0 and later, if you delete a public group, it will be deleted even if it has been used in sharing, consistent with the behavior for UserRole. In versions before 13.0, such sharing prevents the record from being deleted.

SEE ALSO:

[GroupMember](#)

[Object Basics](#)

GroupMember

Represents a User or Group that is a member of a public group.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users cannot access this object.

Fields

Field	Details
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Group.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the User or Group that is a direct member of the group.</p>

Usage

A record exists for every User or Group who is a direct member of a public group whose `Type` field is set to Regular. User records that are indirect members of Regular public groups are not listed as group members. A User can be an indirect member of a group if he or she is in a UserRole above the direct group member in the hierarchy, or if he or she is a member of a group that is included as a subgroup in that group.

If you attempt to create a record that matches an existing record, system simply returns the existing record.

SEE ALSO:

[Object Basics](#)

HashtagDefinition

HashtagDefinition represents hashtag (#) topics in public Chatter posts and comments. Public posts and comments include those on profiles and in public groups, but not those on records or in private groups. This object is available in API version 26.0 and later.

! **Important:** Starting in Spring '16, API access to HashtagDefinition is disabled across all API versions. Any integrations relying on API queries to this object stop working. You can continue to use hashtags in posts and comments, and the hashtags continue to create corresponding topics. We recommend that you redirect all API queries and reports using the HashtagDefinition object to use the Topic object instead. For more information, see [Retiring the Legacy HashtagDefinition Object—FAQs](#).

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
HashtagCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of times a hashtag topic is used.</p>
Name	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The string of characters following the hashtag (#) in a hashtag topic.</p>
NameNorm	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The string of characters following the hashtag (#) in a hashtag topic, normalized to remove capitalization and punctuation.</p>
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Identifier of the community to which the HashtagDefinition belongs. This field is available only if Salesforce Communities is enabled in your organization.</p>

Usage

Use this object to identify public hashtag topics and see how often they're used.

SEE ALSO:

[Topic](#)

Holiday

Represents a period of time during which your customer support team is unavailable. Business hours and escalation rules associated with business hours are suspended during any holidays with which they are affiliated.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

All users, even those without the "View Setup and Configuration" user permission, can view holidays via the API.

Fields

Field	Details
ActivityDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description If the Holiday <code>IsAllDay</code> flag is set to <code>true</code> (indicating that it is an all-day holiday), then the holiday due date information is contained in the <code>ActivityDate</code> field. This field is a date field with a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. The timestamp is not relevant, and you should not attempt to alter it to account for any time zone differences.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of the holiday.</p>
EndTimeInMinutes	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The end time of the holiday in minutes.</p>

Field	Details
IsAllDay	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the duration of the holiday is all day (<code>true</code>) or not (<code>false</code>).</p>
IsRecurrence	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the holiday is scheduled to repeat itself (<code>true</code>) or only occurs once (<code>false</code>). This is a read only field on update, but not on create. If this field value is <code>true</code>, then any recurrence fields associated with the given recurrence type must be populated.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the holiday.</p>
RecurrenceDayOfMonth	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The day of the month on which the holiday repeats.</p>
RecurrenceDayOfWeekMask	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The day or days of the week on which the holiday repeats. This field contains a bitmask. For each day of the week, the values are as follows:</p> <ul style="list-style-type: none"> • Sunday = 1 • Monday = 2 • Tuesday = 4 • Wednesday = 8

Field	Details
	<ul style="list-style-type: none"> • Thursday = 16 • Friday = 32 • Saturday = 64
	<p>Multiple days are represented as the sum of their numerical values. For example, Tuesday and Thursday = 4 + 16 = 20.</p>
RecurrenceEndDateOnly	Type
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The last date on which the holiday repeats. For multiday recurring events, this is the day on which the last occurrence starts.</p>
RecurrenceInstance	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	<p>The frequency of the recurring holiday. For example, 2nd or 3rd.</p>
RecurrenceInterval	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The interval between recurring holidays.</p>
RecurrenceMonthOfYear	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	<p>The month of the year on which the event repeats.</p>
RecurrenceStartDate	Type
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The date when the recurring holiday begins. Must be a date and time before RecurrenceEndDateOnly.</p>

Field	Details
RecurrenceType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates how often the holiday repeats. For example, daily, weekly, or every Nth month (where "Nth" is defined in <code>RecurrenceInstance</code>).</p>
StartTimeInMinutes	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The start time of the holiday in minutes.</p>

Usage

Use this object to view and update holidays, which specify dates and times at which associated business hours and escalation rules are suspended.

Idea

Represents an idea on which users are allowed to comment and vote, for example, a suggestion for an enhancement to an existing product or process. This object is available in API version 12 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

 **Note:** For other standard objects, the `describeLayout()` call returns the `recordTypeMappings` section that contains the layout ID and picklist values for each record type. However, the `recordTypeMappings` section and the fields it includes are not available for the Idea object.

When performing a SOSL search on Idea objects, IdeaComment objects are also searched.

Fields

Field	Details
AttachmentBody	<p>Type base64</p>

Field	Details
	<p>Properties Create, Nillable, Update</p> <p>Description File data for the attachment. This field is available in API version 28.0 and later.</p>
AttachmentContentType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Type of the attachment. This field is available in API version 28.0 and later.</p>
AttachmentLength	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the attachment in bytes. This field is available in API version 28.0 and later.</p>
AttachmentName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name of the attachment. This field is available in API version 28.0 and later.</p>
Body	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the Idea.</p>
Categories	<p>Type multipicklist</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Customizable multi-select picklist used to organize Ideas into logical groupings.</p>

Field	Details
	<p> Note: This field is only available if your organization has the <code>Categories</code> field enabled. This field is enabled by default in organizations created after API version 14 was released. If the <code>Categories</code> field is enabled, API versions 13 and earlier do not have access to either the <code>Categories</code> or <code>Category</code> fields.</p>
Category	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Customizable picklist of values used to organize Ideas into logical groupings.</p> <p> Note: This field is not available if your organization has the multi-select <code>Categories</code> field enabled.</p>
CommunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The zone ID associated with the idea. Once you create an idea, you can't change the zone ID associated with that idea.</p> <p> Note: API version 12 does not support zone ID. If you create an idea in version 12, your idea is automatically posted to the oldest zone that you have permission to access.</p>
CreatorFullPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL of the user's profile photo. This field is available in API version 28.0 and later.</p>
CreatorName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the user who posted the idea or commented on the idea. This field is available in API version 28.0 and later.</p>

Field	Details
CreatorSmallPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL of the user's thumbnail photo. This field is available in API version 28.0 and later.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
IdeaThemeID	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Identifies the idea theme associated with the idea.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsHtml	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. If this value is <code>true</code>, your organization has the Ideas HTML editor enabled, and the Idea Body may contain HTML. If this value is <code>false</code>, the HTML editor is disabled and the Idea Body only contains regular text.</p>
IsMerged	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read only. Indicates whether the idea has been merged with a parent idea (<code>true</code>) or not (<code>false</code>). You can't vote for or add comments to a merged idea.</p> <p> Note: In API version 27, <code>IsMerged</code> replaces <code>IsLocked</code>. Existing formula fields that use <code>IsLocked</code> must be edited to use <code>IsMerged</code>.</p>
LastCommentDate	<p>Type <code>dateTime</code></p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time the last comment (child <code>IdeaComment</code> object) was added.</p>
LastCommentId	<p>Type <code>reference</code></p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read only. The ID of the last comment (child <code>IdeaComment</code> object).</p>
LastReferencedDate	<p>Type <code>date</code></p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type <code>date</code></p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
NumComments	<p>Type <code>int</code></p> <p>Properties Defaulted on create, Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The number of comments (child IdeaComment objects) that users have submitted for the given idea.</p>
ParentIdeaId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID associated with this idea's parent idea. When multiple ideas are merged together, one idea becomes the parent (master) of the other ideas. The ParentIdeaId is automatically set when you merge ideas.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the record type assigned to this object.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Customizable picklist of values used to specify the status of an idea.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The descriptive title of the idea.</p>
VoteScore	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The internal score of the Idea, used to sort Ideas on the Popular tab in the application user interface. The internal algorithm that determines the score gives older votes less weight than</p>

Field	Details
	<p>newer votes, simulating exponential decay. The score itself does not display in the application user interface.</p> <p> Note: Unlike other fields of type double, you can't use a SOQL aggregate function with this field.</p>
VoteTotal	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description An Idea's total number of points. Each vote a user makes is worth ten points, therefore the value of this field is ten times the number of votes an idea has received.</p> <p> Note: Unlike other fields of type double, you can't use a SOQL aggregate function with this field.</p>

 **Note:** If you are importing Idea data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Usage

Use this object to track ideas, which are written suggestions on which users can vote and comment.

SEE ALSO:

[IdeaComment](#)

[Vote](#)

IdeaComment

Represents a comment that a user has submitted in response to an idea.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

 **Note:** When performing a SOSL search on IdeaComment objects, Idea objects are also searched.

Fields

Field	Field Type
CommentBody	<p>Type textarea</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Body of the submitted comment.</p>
CommunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The zone ID associated with the idea. Once you create an idea, you can't change the zone ID associated with that idea.</p> <p> Note: API version 12 does not support zone ID. If you create an idea in version 12, your idea is automatically posted to the oldest zone that you have permission to access.</p>
CreatorFullPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description URL of the user's profile photo. This field is available in API version 28.0 and later.</p>
CreatorName	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Name of the user who posted the idea or commented on the idea. This field is available in API version 28.0 and later.</p>
CreatorSmallPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description URL of the user's thumbnail photo. This field is available in API version 28.0 and later.</p>

Field	Field Type
IdeaId	Type reference Properties Create, Filter, Group, Sort Description ID of the idea on which this comment was made.
IsHtml	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Read-only. If this value is <code>true</code> , your organization has the Ideas HTML editor enabled, and the <code>CommentBody</code> field may contain HTML. If this value is <code>false</code> , the HTML editor is disabled and the <code>CommentBody</code> field only contains regular text.
UpVotes	Type int Properties Filter, Group, Nillable, Sort Description Total number of up votes for the question.



Note: If you import these records, and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself.

Usage

Use this object to track comments on ideas, which are users' text responses to ideas.

SEE ALSO:

[Idea](#)
[Vote](#)

IdeaReputation

Represents a collection of statistics and scores derived from a user's activity within an Ideas zone or internal organization. This object is available in API version 28.0 and later.

Supported Calls

`query()`, `retrieve()`,

Fields

Field	Details
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of comments a user has created in a zone or the internal organization. This number excludes comments the user creates on his or her own idea.</p>
CommentsReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of comments a user has received in a zone or the internal organization.</p>
ContextId	<p>Type reference</p> <p>Properties Filter, Group, Namepointing, Nillable, Sort</p> <p>Description The ID of the zone or internal organization.</p>
DownVotesGivenCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of down votes a user has given in a zone or the internal organization.</p>
DownVotesReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of down votes a user has received in a zone or the internal organization.</p>

Field	Details
IdeaCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of ideas a user has created in a zone or the internal organization.</p>
ReputationLevel	<p>Type string</p> <p>Properties Nillable</p> <p>Description The reputation level that a user has achieved based on their score in a zone or within an organization.</p>
Score	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The total score of a user's activity within a zone or within an organization.</p>
UpVotesGivenCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of up votes a user has given in a zone or the internal organization. This number doesn't include the default vote the system applies when the user creates the idea.</p>
UpVotesReceivedCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of up votes a user has received in a zone or the internal organization.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description The user ID associated with the reputation.</p>

Usage

Use to query a user's reputation within a zone.

IdeaReputationLevel

Represents a reputation level within an Ideas zone or internal organization and is used by the system to calculate reputation. You can create up to 25 levels per zone or internal organization. This object is available in API version 28.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ContextId	<p>Type reference</p> <p>Properties Create, Filter, Group, Namepointing, Sort, Update</p> <p>Description The ID of the zone or internal organization.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the reputation level. The name must be unique within the zone or internal organization. Maximum size is 50 characters.</p>
Threshold	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	<p>Minimum number of points for this level. The threshold must be unique within the zone or internal organization and must be greater than or equal to zero.</p>

Usage

Use to create or edit reputation levels for an Ideas zone or internal organization.

IdeaTheme

Represents an invitation to community members to submit ideas that are focused on a specific topic. This object is available in API version 26 and later.

Supported Calls

`create(), delete(), describeLayout(), query(), retrieve(), search(), undelete(), update()`

Fields

Field Name	Details
Categories	<p>Type multipicklist</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Customizable multi-select picklist used to organize ideas and idea themes into logical groupings.</p>
CommunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort,</p> <p>Description The zone ID associated with the idea theme.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nullable, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the idea theme.</p>
EndDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Date marking the end of the idea theme.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
StartDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Date that the idea theme begins.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description Customizable picklist of values used to specify the status of the idea theme.</p>

Field Name	Details
Title	Type string Properties Create, Filter, Group, idLookup, Namefield, Sort, Update Description Title of the idea theme.

Usage

Use the object to track ideas that are submitted to an idea theme.

KnowledgeableUser

Represents a user identified as knowledgeable about a specific topic, and ranks them relative to other knowledgeable users. This object is available in API version 31.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
NetworkId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the community the topic exists in. This field is available only if Salesforce Communities is enabled for your organization.
RawRank	Type int Properties Filter, Group, Nillable, Sort Description Rank of this user's knowledge on the topic relative to other users.

Field Name	Details
TopicId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Unique ID for the topic in Salesforce.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Unique ID for the user in Salesforce.</p>

KnowledgeArticle

Provides read-only access to an article and the ability to delete the master article. This object is available in API version 19.0 and later. Unlike KnowledgeArticleVersion, the ID of a KnowledgeArticle record is identical irrespective of the article's version (status). For more information on articles and article types, see "Work with Articles and Translations" and "Knowledge Article Types" in the Salesforce online help.

Supported Calls

`delete()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

A user must have the "View Articles" permission enabled. Salesforce Knowledge users, unlike Customer Portal and partner portal users, must also be granted the `Knowledge User` feature license.

Fields

Field Name	Details
ArchivedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>

Field Name	Details
ArchivedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date when the article was archived.
ArticleNumber	Type string
	Properties Autonumber, Defaulted on create, Filter, Sort
	Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.
CaseAssociationCount	Type int
	Properties Filter, Group, Sort
	Description The number of cases attached to the article.
FirstPublishedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date when the article was first published.
LastPublishedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date when the article was last published.
LastReferencedDate	Type date
	Properties Filter, Nillable, Sort, Update

Field Name	Details
	Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
MasterLanguage	Type picklist Properties Filter, Restricted picklist Description The article's original language. Only accessible if your knowledge base supports multiple languages.

Usage

Use this object to query or retrieve articles. KnowledgeArticle can be used in a SOQL clause, but doesn't provide access to the fields from the article. Provides read-only access to an article and the ability to delete the master article.

SEE ALSO:

- [Articles](#)
- [Data Categories](#)
- [KnowledgeArticleVersion](#)

KnowledgeArticleVersion

Provides a global view of standard article fields across all article types depending on their version. This object is available in API version 18.0 and later.

Use this object to:

- Query or search generically across multiple article types.
- Filter on a specific version.
- Update standard fields, those fields that are updateable, in draft versions.

When you query on the archived article, the results include both the article and the article's archived versions.

 **Note:** You can't update draft translations with the API.

Supported Calls

`create(), delete(), describeLayout(), describeObjects(), query(), retrieve(), search(), and update().`

 **Note:**

- You can only update draft versions.
- To create, update, or delete a Knowledge article version, use the call on `ArticleType_kav`, where `ArticleType` is the name of the article's type. For example, to delete, use `ArticleType_kav.delete()`.

Special Access Rules

A user must have the “View Articles” permission enabled. Salesforce Knowledge users, unlike Customer Portal and partner portal users, must also be granted the `Knowledge User` feature license to view and edit article versions.

Fields

Field Name	Details
ArchivedById	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description The ID of the user who archived the article.</p>
ArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description The date when the article was archived.</p>
ArticleNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter</p> <p>Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.</p>
ArticleType	<p>Type string</p>

Field Name	Details
	<p>Properties Defaulted on create,Filter</p> <p>Description Indicates the API Name of the article type. The <code>ArticleType</code> is assigned to the article when it's created. You can't change the value of this field. This field is available in API version 26.0 and later.</p>
FirstPublishedDate	<p>Type <code>dateTime</code></p> <p>Properties Filter, Nillable</p> <p>Description The date when the article was first published.</p>
IsLatestVersion	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Group, Filter, Sort</p> <p>Description Indicates whether the article is the most current version. (<code>true</code>) or not (<code>false</code>). This field can be <code>true</code> on the online or published version, a draft version in the master language, a draft version in a translation, and the latest archived version. However, you can't filter by (<code>PublishState='Online'</code>) and (<code>IsLatestVersion=false</code>) because the online version is also the latest version. This field is available in API version 24.0 and later.</p>
IsMasterLanguage	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the article has one or more translations associated with it (<code>true</code>) or not (<code>false</code>). Only accessible if your knowledge base supports multiple languages.</p>
IsOutOfDate	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the source article has been updated since this translated version was created (<code>true</code>) or not (<code>false</code>). Only accessible if your knowledge base supports multiple languages.</p>

Field Name	Details
IsVisibleInApp	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Required. Indicates whether the article is visible in the Articles tab (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInCsp	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Required. Indicates whether the article is visible in the Customer Portal (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPkb	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Required. Indicates whether the article is visible in the public knowledge base (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPrm	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Required. Indicates whether the article is visible in the partner portal (<code>true</code>) or not (<code>false</code>).</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Filter</p> <p>Description The ID of the article independent from its version. The value for this field is retrieved from the <code>Id</code> field of the KnowledgeArticle object.</p>
Language	<p>Type picklist</p>

Field Name	Details
	<p>Properties Filter, Restricted picklist</p>
	<p>Description The language that the article is written in, such as French or Chinese (Traditional). Querying or searching articles in SOQL or SOSL requires that you specify the Language field in the WHERE clause. The language must be the same for all article types.</p>
LastPublishedDate	<p>Type dateTime</p>
	<p>Properties Filter, Nillable</p>
	<p>Description The date when the article was last published.</p>
MasterVersionId	<p>Type reference</p>
	<p>Properties Filter, Nillable</p>
	<p>Description ID of the source article, if the article is the translation of a source article. Only accessible if your knowledge base supports multiple languages.</p>
OwnerId	<p>Type reference</p>
	<p>Properties Filter</p>
	<p>Description The ID of the article's owner.</p>
PublishStatus	<p>Type picklist</p>
	<p>Properties Defaulted on create, Filter, Restricted picklist</p>
	<p>Description The publication status for the article: <ul style="list-style-type: none"> • Draft: any draft articles. • Online: articles published in Salesforce Knowledge. • Archived: archived articles. A user must have the "Manage Articles" permission enabled to use Online. </p>

Field Name	Details
	<p>Querying or searching articles in SOQL or SOSL requires that you specify either the <code>PublishStatus</code> or the <code>Id</code> field in the WHERE clause. You can search for only one publication status per article type in a single SOSL query. When searching for articles with a <code>PublishStatus</code> of <code>Archived</code>, also check that <code>IsLatestVersion</code> equals <code>false</code> in your WHERE clause.</p>
SourceId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the source from which the article was created (Case or Reply).</p>
Summary	<p>Type textarea</p> <p>Properties Filter, Nillable</p> <p>Description Summary of the article. Maximum size is 1000 characters.</p>
Title	<p>Type string</p> <p>Properties Defaulted on create, Filter, idLookup</p> <p>Description Required. Article's title. Maximum size is 255 characters.</p>
TranslationCompletedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description Date and time when the article was last translated. Only accessible if your knowledge base supports multiple languages.</p>
TranslationExportedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description Date and time when the article was last exported for translation. Only accessible if your knowledge base supports multiple languages.</p>

Field Name	Details
TranslationImportedDate	Type dateTime
	Properties Filter, Nillable
	Description
	Date and time when the article was last imported for translation. Only accessible if your knowledge base supports multiple languages.
UrlName	Type string
	Properties Filter
	Description
	Required. Represents the article's URL. Can contain alphanumeric characters and hyphens but can't begin or end with a hyphen. <code>UrlName</code> is case-sensitive and its maximum size is 255 characters.
ValidationStatus	Type picklist
	Properties Filter, Group
	Description
	Shows whether the content of the article has been validated. Default values are <code>Validated</code> and <code>Not Validated</code> . This field is available in API version 24.0 or later.
VersionNumber	Type int
	Properties None
	Description
	The number assigned to a version of an article. This field is available in API version 24.0 and later.

Usage

Use this object to query, retrieve, or search for articles across all article types depending on their version. You can update draft master articles. Additionally, you can delete articles that are not drafts. Client applications can use `KnowledgeArticleVersion` with `describeDataCategoryGroups()` and `describeDataCategoryGroupStructures()` to return the category groups and the category structure associated with Salesforce Knowledge.

Alternatively, client applications can use the article type `API Name` followed by `__kav` to query, retrieve, or search for articles for a specific article type.

To access an article independent of its version, use the KnowledgeArticle object.

SOQL Samples

The following SOQL clause uses KnowledgeArticleVersion to query all published articles from **all article types** complying with the classification specified in the WITH DATA CATEGORY clause:

```
SELECT Title, Summary
FROM KnowledgeArticleVersion
WHERE PublishStatus='Online'
AND Language = 'en_US'
WITH DATA CATEGORY Geography__c ABOVE_OR_BELOW europe__c AND Product__c BELOW All__c
```

The following SOQL clause uses the article type API Name to limit the query to all draft articles from the offer__kav article type:

```
SELECT Id, Title
FROM Offer__kav
WHERE PublishStatus='Draft'
AND Language = 'en_US'
WITH DATA CATEGORY Geography__c AT (france__c,usa__c) AND Product__c ABOVE dsl__c
```

The following SOQL clause uses KnowledgeArticleVersion to query the Ids of all archived versions of a particular article:

```
SELECT Id
FROM KnowledgeArticleVersion
WHERE PublishStatus='Archived'
AND IsLatestVersion=false
AND KnowledgeArticleId='kA1D00000001PQ6KAM'
```

SOQL and SOSL with KnowledgeArticleVersion

- Always filter on a single value of PublishStatus unless the query filters on one or more primary key IDs. To support security, only users with the “Manage Articles” permission see articles whose PublishStatus value is Draft.
- Archived article versions are stored in the articletype_kav object. To query archived article versions, specify the article Id and set IsLatestVersion='0'.
- Always filter on a single value of Language. However, in SOQL, you can filter on more than one Language if there is a filter on Id or KnowledgeArticleId.

SEE ALSO:

[Articles](#)
[Data Categories](#)
[KnowledgeArticle](#)
[KnowledgeArticleViewStat](#)
[KnowledgeArticleVoteStat](#)

KnowledgeArticleVersionHistory

Enables read-only access to the full history of an article. This object is available in API version 25.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

This object respects field, entity, and record-level security. You must have at least “Read” permission on the article type or the field to access its history. For data category security, Salesforce determines access based on the categorization of the online version of an article. If there is no online version, then security is applied based on the archived version, followed by the security of the draft version.

Fields

Field Name	Details
EventType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of event that is tracked in the history table.</p>
FieldName	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Name of the tracked field.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language that the article is written in, such as French or Chinese (Traditional). Querying or searching articles in SOQL or SOSL requires that you specify the Language field in the WHERE clause. The language must be the same for all article types.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	<p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The most recent value of the field before it was changed.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the article.</p>
ParentObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of object that contains the field.</p>
VersionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID assigned to a version of the article.</p>
VersionNumber	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>

Usage

Use this object to query events in the history of an article. For example, you can retrieve the number of edits a particular user has made to an article, how many times the article has been published, and so on.

KnowledgeArticleViewStat

Provides statistics on the number of views for the specified article across all article types. This object is read-only and available in API version 20 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Users must have access to the published version of an article to retrieve its views. For more information on published article version, see the `PublishStatus` field in `KnowledgeArticleVersion`.

Fields

Field Name	Details
Channel	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The channel where the article is viewed:</p> <ul style="list-style-type: none">• <code>AllChannels</code> for article views across all channels.• <code>App</code> for the internal Salesforce Knowledge application.• <code>Pkb</code> for article views in public knowledge base.• <code>Csp</code> for Customer Portal.• <code>Prm</code> for article view in partner portal.
NormalizedScore	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description Article's weighted views in the selected channel. The article with most views has a score of 100. Other article views are then calculated relative to this highest view score. For example, if the best read article has 2000 views and another has 1000. The first one gets a score of 100 while the second gets 50.</p>

Field Name	Details
ParentId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the viewed article. This corresponds to a KnowledgeArticle record.</p>
ViewCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of unique views an article has received in the selected channel. An article with a high number of views may not always have a high normalized score. The normalized score for an article is calculated based on views over time, with more recent views earning a higher score. This field is available in API version 27.0 and later.</p>

Usage

Use this object to query or retrieve statistics for article views.

Alternatively, client applications can use the article type API Name followed by __ViewStat to query or retrieve most viewed articles from a specific article type.

SOQL Samples

The following SOQL clause uses KnowledgeArticleViewStat to query all the article views in Salesforce Knowledge and return the related articles:

```
SELECT Id, NormalizedScore, Parent.Id
      FROM KnowledgeArticleViewStat WHERE Channel = 'App'
      ORDER BY NormalizedScore
```

Use the following clause to restrict your query to Offer articles for the `Offer` article type:

```
SELECT Id, NormalizedScore, Parent.Id  
      FROM Offer__ViewStat WHERE Channel = 'App'  
      ORDER BY NormalizedScore
```

SEE ALSO:

[Articles](#)
[Data Categories](#)
[KnowledgeArticle](#)
[KnowledgeArticleVersion](#)
[KnowledgeArticleVoteStat](#)

KnowledgeArticleVoteStat

Provides the weighted rating for the specified article on a scale of 1 to 5 across all article types. This object is read-only and available in API version 20 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Users must have access to the published version of an article to retrieve its votes. For more information on published article version, see the `PublishStatus` field in `KnowledgeArticleVersion`

Fields

Field Name	Details
Channel	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The channel where the article is rated:<ul style="list-style-type: none">• <code>AllChannels</code> for article views across all channels.• <code>App</code> for the internal Salesforce Knowledge application.• <code>Pub</code> for article views in public knowledge base.• <code>Csp</code> for Customer Portal.• <code>Prm</code> for article view in partner portal.</p>

Field Name	Details
NormalizedScore	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description Article's weighted score on a scale of 1 to 5. A higher score means more votes. Articles without recent votes trend towards an average rating of three stars.</p>
ParentId	<p>Type reference</p> <p>Properties Filter</p> <p>Description The rated article. This corresponds to a KnowledgeArticle record.</p>

Usage

Use this object to query or retrieve the rating for an article.

Alternatively, client applications can use the article type API Name followed by `__VoteStat` to query or retrieve the rating for an article for a specific article type.

SOQL Samples

See `KnowledgeArticleViewStat`.

SEE ALSO:

- [Articles](#)
- [Data Categories](#)
- [KnowledgeArticle](#)
- [KnowledgeArticleVersion](#)
- [KnowledgeArticleViewStat](#)

Lead

Represents a prospect or potential Opportunity.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `merge()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Address	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
AnnualRevenue	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Annual revenue for the company of the lead.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description City for the address of the lead.</p>
CleanStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates the record's clean status as compared with Data.com. Values are: Matched, Different, Acknowledged, NotFound, Inactive, Pending, SelectMatch, or Skipped. Several values for CleanStatus display with different labels on the lead record detail page. <ul style="list-style-type: none"> • Matched displays as In Sync • Acknowledged displays as Reviewed • Pending displays as Not Compared </p>
Company	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Company of the lead.</p> <p> Note: If person account record types have been enabled, and if the value of Company is null, the lead converts to a person account.</p>
CompanyDunsNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun&Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used by industries and organizations around the world as a global standard for business identification and tracking. Maximum size is 9 characters.</p> <p> Note: This field is only available to organizations that use Data.com Prospector or Data.com Clean.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
ConvertedAccountId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Object reference ID that points to the Account into which the Lead has been converted.</p>
ConvertedContactId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Object reference ID that points to the Contact into which the Lead has been converted.</p>
ConvertedDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Date on which this Lead was converted.</p>
ConvertedOpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Object reference ID that points to the Opportunity into which the Lead has been converted.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Country for the address of the lead.</p>
CountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the lead's address.</p>
CurrencyIsoCode	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the lead.</p>
Division	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.</p>
Email	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Email address for the lead.</p>
EmailBouncedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description If bounce management is activated and an email sent to the lead bounced, the date and time the bounce occurred.</p>
EmailBouncedReason	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description If bounce management is activated and an email sent to the lead bounced, the reason the bounce occurred.</p>
Fax	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Fax number for the lead.</p>
FirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description First name of the lead. Limited to 40 characters.</p>
HasOptedOutOfEmail	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the lead has opted out of email (<code>true</code>) or not (<code>false</code>). Label is Email Opt Out.</p>
GeocodeAccuracy	<p>Type picklist</p> <p>Properties Retrieve, Query, Restricted picklist, Nillable</p> <p>Description Accuracy level of the geocode for the address on this object. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
Industry	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Industry the lead works in.</p>

Field	Details
IsConverted	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the Lead has been converted (<code>true</code>) or not (<code>false</code>). Label is Converted.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsUnicodeedByOwner	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If true, lead has been assigned, but not yet viewed. See Unread Leads for more information. Label is Unread By Owner.</p>
Jigsaw	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description References the ID of a contact in Data.com. If a lead has a value in this field, it means that a contact was imported as a lead from Data.com. If the contact (converted to a lead) was not imported from Data.com, the field value is null. Maximum size is 20 characters. Available in API version 22.0 and later. Label is Data.com Key.</p> <p> Important: The Jigsaw field is exposed in the API to support troubleshooting for import errors and reimporting of corrected data. Do not modify the value in the Jigsaw field.</p>
LastActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description</p> <p>Value is one of the following, whichever is the most recent:</p> <ul style="list-style-type: none">• Due date of the most recent event logged against the record.• Due date of the most recently closed task associated with the record.
LastName	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Sort, Update</p> <p>Description</p> <p>Required. Last name of the lead. Limited to 80 characters.</p>
LastReferencedDate	<p>Type</p> <p>date</p> <p>Properties</p> <p>Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type</p> <p>date</p> <p>Properties</p> <p>Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Latitude	<p>Type</p> <p>double</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>Used with <code>Longitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
Longitude	<p>Type</p> <p>double</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description Used with <code>Latitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
LeadSource	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Source from which the lead was obtained.</p>
MasterRecordId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If this object was deleted as the result of a merge, this field contains the ID of the record that was kept. If this object was deleted for any other reason, or has not been deleted, the value is <code>null</code>.</p> <p> Note: When using Apex triggers to determine which record was deleted in a merge event, this field's value is the ID of the record that was kept only in <code>Trigger.old</code>. In <code>Trigger.new</code>, the value is <code>null</code>.</p>
MiddleName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Middle name of the lead. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
MobilePhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Mobile phone number for the lead.</p>
Name	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description Concatenation of FirstName, MiddleName, LastName, and Suffix. Maximum size is 121 characters.</p>
NumberOfEmployees	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Number of employees at the lead's company. Label is Employees.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the lead.</p>
PartnerAccountId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the partner account for the partner user that owns this lead. Available only if Partner Relationship Management is enabled OR Communities is enabled and you have partner portal licenses.</p> <p> Note: If you are uploading leads using API version 15.0 or earlier, and one of the leads in the batch has a partner user as the owner, the Partner Account field on all leads in the batch is set to that partner user's account regardless of whether the partner user is the owner. In version 16.0, the Partner Account field is set to the appropriate account for the partner user that owns the lead. If the owner of the lead is not a partner user, this field remains blank.</p>
Phone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Phone number for the lead.</p>

Field	Details
PhotoUrl	<p>Type url</p> <p>Properties Filter, Group, Sort</p> <p>Description Path to be combined with the URL of a Salesforce instance (for example, https://yourInstance.salesforce.com/) to generate a URL to request the social network profile image associated with the lead. Generated URL returns an HTTP redirect (code 302) to the social network profile image for the lead.</p> <p>Blank if Social Accounts and Contacts isn't enabled for the organization or if Social Accounts and Contacts has been disabled for the requesting user.</p>
PostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal code for the address of the lead. Label is Zip/Postal Code.</p>
Rating	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Rating of the lead.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
Salutation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Salutation for the lead.</p>

Field	Details
State	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the address of the lead.</p>
StateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the lead's address.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Status code for this converted lead. Status codes are defined in <code>status</code> and represented in the API by the LeadStatus object.</p>
Street	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street number and name for the address of the lead.</p>
Suffix	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name suffix of the lead. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	Description
Website	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Website for the lead.</p>

 **Note:** If you import Lead data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Converted Leads

Leads have a special state to indicate that they have been converted into an Account, Contact, and optionally, an Opportunity. Your client application can convert leads via the `convertLead()` call. Users can also convert leads through the user interface. Once a lead has been converted, it is read-only. You can't update or delete a converted lead. However, you can query converted lead records.

Leads have several fields that indicate their converted status. These special fields are set when converting the lead in the user interface.

- `ConvertedAccountId`
- `ConvertedContactId`
- `ConvertedDate`
- `ConvertedOpportunityId`
- `IsConverted`
- `Status`

 **Note:** If person account record types have been enabled, and if the value of `Company` is null, the lead converts to a person account.

Unread Leads

Leads have a special state to indicate that they have not been viewed or edited by the lead owner. In the user interface, this is helpful for users to know which leads have been assigned to them but which they have not touched yet. `IsUnreadByOwner` is `true` if the lead owner has not yet viewed or edited the lead, and `false` if the lead owner has viewed or edited the lead at least once.

Lead Status Picklist

Each `Status` value corresponds to either a converted or unconverted status in the lead status picklist, as defined in the user interface. To obtain the lead status values in the picklist, a client application can query `LeadStatus`.

You can't convert a lead via the API by changing `Status` to one of the converted lead status values. When you convert qualified leads into an account, contact, and opportunity, you can select one of the converted status types for the lead. Leads with a converted status type are no longer available in the Leads tab, although you can include them in reports.

Usage

To update a Lead or to convert one with `convertLead()`, your client application must log in with the “Edit” permission on leads.

When you create, update, or upsert a lead, your client application can have the lead automatically assigned to one or more User records based on assignment rules that have been configured in the user interface.

To use this feature, your client application needs to set either of the following options (but not both) in the AssignmentRuleHeader used in create or update:

Field	Field Type	Details
<code>assignmentRuleId</code>	reference	ID of the assignment rule to use. Can be an inactive assignment rule. If unspecified and <code>useDefaultRule</code> is <code>true</code> , then the default assignment rule is used. To find the ID for a given assignment rule, query the AssignmentRule object (specifying <code>RuleType="leadAssignment"</code>), iterate through the returned AssignmentRule records, find the one you want to use, retrieve its ID, and then specify its ID in this field in the AssignmentRuleHeader.
<code>useDefaultRule</code>	boolean	Specifies whether to use the default rule for rule-based assignment (<code>true</code>) or not (<code>false</code>). Default rules are assigned in the user interface.

Java Sample

The following Java sample shows how to automatically assign a newly created lead.

```
package wsc;

import com.sforce.soap.enterprise.Connector;
import com.sforce.soap.enterprise.EnterpriseConnection;
import com.sforce.ws.ConnectionException;
import com.sforce.ws.ConnectorConfig;
import com.sforce.soap.enterprise.sobject.Lead;
import com.sforce.soap.enterprise.QueryResult;
import com.sforce.soap.enterprise.SaveResult;
import com.sforce.soap.enterprise.sobject.SObject;

public class LeadAssignment {

    static final String USERNAME = "REPLACE USER NAME";
    static final String PASSWORD = "REPLACE PASSWORD";
    static EnterpriseConnection connection;

    static LeadAssignment _leadAssignment;

    // Main
    public static void main(String[] args)
    {
        // Establish connection and login
        ConnectorConfig config = new ConnectorConfig();
        config.setUsername(USERNAME);
        config.setPassword(PASSWORD);
        try {

```

```
connection = Connector.newConnection(config);
System.out.println("Logged in, endpoint: " + config.getAuthEndpoint());
} catch (ConnectionException e1) {
    e1.printStackTrace();
}

// Create lead
_leadAssignment = new LeadAssignment();
try {
    _leadAssignment.CreateLead();
} catch (Exception e) {
    e.printStackTrace();
}

// Logout
try {
    connection.logout();
    System.out.println("Logged out");
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

public void CreateLead() throws ConnectionException
{
    // Create a new Lead and assign various properties
    Lead lead = new Lead();

    lead.setFirstName("Joe");
    lead.setLastName("Smith");
    lead.setCompany("ABC Corporation");
    lead.setLeadSource("API");
    // The lead assignment rule will assign any new leads that
    // have "API" as the LeadSource to a particular user

    // In this sample we will look for a particular rule and if found
    // use the id for the lead assignment. If it is not found we will
    // instruct the call to use the current default rule. You can't use
    // both of these values together.
    QueryResult qr = connection.query("SELECT Id FROM AssignmentRule WHERE Name = " +
        "'Mass Mail Campaign' AND SObjectType = 'Lead'");
    if (qr.getSize() == 0) {
        connection.setAssignmentRuleHeader(null, true);
    } else {
        connection.setAssignmentRuleHeader(qr.getRecords()[0].getId(), false);
    }

    // Every operation that results in a new or updated lead will
    // use the specified rule until the header is removed from the
    // connection.
    SaveResult[] sr = connection.create(new SObject[] {lead});
    for (int i=0;i<sr.length;i++) {
        if (sr[i].isSuccess()) {
```

```
        System.out.println("Successfully created lead with id of: " +
                           sr[i].getId() + ".");
    } else {
        System.out.println("Error creating lead: " +
                           sr[i].getErrors()[0].getMessage());
    }
}

// This call effectively removes the header, the next lead will
// be assigned to the default lead owner.
connection.clearAssignmentRuleHeader();
}
}
```

C# Sample

The following C# sample shows how to automatically assign a newly created lead.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.ServiceModel;
using LeadSample.sforce;

namespace LeadSample
{
    class LeadAssignment
    {
        private static SoapClient client;
        private static SoapClient apiClient;
        private static SessionHeader header;
        private static LoginResult loginResult;
        private static readonly string Username = "REPLACE USERNAME";
        private static readonly string Password = "REPLACE PASSWORD AND SECURITY TOKEN";

        // Create the proxy binding and login
        private LeadAssignment()
        {
            client = new SoapClient();

            try
            {
                loginResult = client.login(null, Username, Password);
            }
            catch (Exception e)
            {
                Console.WriteLine("Unexpected login error: " + e.Message);
                Console.WriteLine(e.StackTrace);
                return;
            }
            // Access API endpoint and create new client
            header = new SessionHeader();
```

```
        header.sessionId = loginResult.sessionId;
        apiClient = new SoapClient("Soap", loginResult.serverUrl);
    }

    [STAThread]
    static void Main(string[] args)
    {
        LeadAssignment leadAssignment = new LeadAssignment();
        try
        {
            leadAssignment.CreateLead();
        }
        catch (Exception e)
        {
            Console.WriteLine(e.Message);
            Console.WriteLine(e.StackTrace);
            Console.WriteLine(e.InnerException);
        }
        // logout
        client.logout(header);
    }

    public void CreateLead()
    {
        // Create a new Lead and assign various properties
        Lead lead = new Lead();

        lead.FirstName = "John";
        lead.LastName = "Brown";
        lead.Company = "ABC Corporation";
        lead.LeadSource = "Advertisement";
        // Setting the lead source for a pre-existing lead assignment rule. This
        // rule was created outside of this sample and will assign any new leads
        // that have "Advertisement" as the LeadSource to a particular user.

        // Create the assignment rule header and add it to the proxy binding
        AssignmentRuleHeader arh = new AssignmentRuleHeader();

        // In this sample we will look for a particular rule and if found
        // use the id for the lead assignment. If it is not found we will
        // instruct the call to use the current default rule. Both these
        // values can't be used together.
        QueryResult qr = null;
        string query = "SELECT Id FROM AssignmentRule WHERE Name = " +
                      "'Mass Mail Campaign' AND SObjectType = 'Lead'";
        try
        {
            LimitInfo[] limitArray = apiClient.query(
                header, // sessionheader
                null, // queryoptions
                null, // mruheader
                null, // packageversionheader
                query, // SOQL query
                out qr);
        }
    }
}
```

```
        }

    catch (Exception e)
    {
        Console.WriteLine("Unexpected query error: " + e.Message);
        Console.WriteLine(e.StackTrace);
    }

    if (qr.size == 0)
    {
        arh.useDefaultRule = true;
    }
    else
    {
        arh.assignmentRuleId = qr.records[0].Id;
    }

    // Create the lead using our Assignment Rule header
    LimitInfo[] li;
    SaveResult[] sr;
    apiClient.create(
        header, // sessionheader
        arh, // assignmentruleheader
        null, // mruheader
        null, // allowfieldtrunctionheader
        null, // disablefeedtrackingheader
        null, // streamingenabledheader
        null, // allornoneheader
        null, // duplicateruleheader
        null, // localeoptions
        null, // debuggingheader
        null, // packageversionheader
        null, // emailheader
        new sObject[] { lead },
        out li,
        out sr);
}

foreach (SaveResult s in sr)
{
    if (s.success)
    {
        Console.WriteLine("Successfully created Lead with ID: {0}", s.id);
    }
    else
    {
        Console.WriteLine("Error creating Lead: {0}", s.errors[0].message);
    }
}
}
```

```
}
```

SEE ALSO:

[LeadOwnerSharingRule](#)
[LeadShare](#)
[LeadStatus](#)
[PartnerNetworkConnection](#)

LeadCleanInfo

Stores the metadata Data.com Clean uses to determine a lead record's clean status. Helps you automate the cleaning or related processing of lead records.

Lead Clean Info provides a snapshot of the data in your Salesforce lead record and its matched Data.com record at the time the Salesforce record was cleaned.

Lead Clean Info includes a number of bit vector fields, whose component fields each correspond to individual object fields and provide related data or status information about those fields. For example, the bit vector field `IsDifferent` has an `IsDifferentTitle` field. If the `IsDifferentTitle` field's value is `False`, that means the `Title` field value is *the same* on the Salesforce lead record and its matched Data.com record.

LeadCleanInfo bit vector fields include:

- `CleanedBy` indicates who (a user) or what (a Clean job) cleaned the lead record.
- `IsDifferent` indicates whether or not a field on the lead record has a value that differs from the corresponding field on the matched Data.com record.
- `IsFlaggedWrong` indicates whether or not a field on the lead record has a value that is flagged as wrong to Data.com.
- `IsReviewed` indicates whether or not a field on the lead record is in a `Reviewed` state, which means that the value was reviewed but not accepted..

Their individual bits are defined here

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
Address	Type address Properties Filter, Nillable

Field Name	Details
	Description
	The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.
AnnualRevenue	Type currency
	Properties
	Filter, Nillable, Sort
	Description
	Estimated annual revenue of the lead.
City	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the lead.
CleanedByJob	Type boolean
	Properties
	Filter
	Description
	Indicates whether the lead record was cleaned by a Data.com Clean job (<code>true</code>) or not (<code>false</code>).
CleanedByUser	Type boolean
	Properties
	Filter
	Description
	Indicates whether the lead record was cleaned by a Salesforce user (<code>true</code>) or not (<code>false</code>).
CompanyDunsNumber	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The Data Universal Numbering System (D-U-N-S) number is a unique, nine-digit number assigned to every business location in the Dun & Bradstreet database that has a unique, separate, and distinct operation. D-U-N-S numbers are used

Field Name	Details
	<p>by industries and organizations around the world as a global standard for business identification and tracking.</p>
CompanyName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the company.</p>
ContactStatusDataDotCom	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The status of the contact associated with the lead per Data.com. Values are: Contact is Active per Data.com, Phone is Wrong per Data.com , Email is Wrong per Data.com, Phone and Email are Wrong per Data.com, Contact Not at Company per Data.com, Contact is Inactive per Data.com, Company this contact belongs to is out of business per Data.com, Company this contact belongs to never existed per Data.com or Email address is invalid per Data.com.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address of the lead.</p>
DandBCompanyDunsNumber	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The D-U-N-S Number on the D&B Company record (if any) that is linked to the lead.</p>
DataDotComCompanyId	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID Data.com maintains for the company associated with the lead.</p>
DataDotComId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID Data.com maintains for the contact associated with the lead.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The lead's email address.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The lead's first name.</p>
Industry	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The industry the lead belongs to.</p>
IsDifferentAnnualRevenue	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>AnnualRevenue</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentCity	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>City</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCompanyDunsNumber	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>Company D-U-N-S Number</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCompanyName	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>Company Name</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountry	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>Country</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentCountryCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>Country Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentDandBCompanyDunsNumber	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's D&B Company D-U-N-S Number field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's Email field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentFirstName	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's First Name field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentIndustry	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's Industry field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentLastName	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's Last Name field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentNumberOfEmployees	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>No. of Employees</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentPhone	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>Phone</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentPostalCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>Postal Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentState	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the lead's <code>State</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>
IsDifferentStateCode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether the account's <code>State Code</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).</p>

Field Name	Details
IsDifferentStreet	Type boolean
	Properties Filter
	Description Indicates whether the lead's <code>Street</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsDifferentTitle	Type boolean
	Properties Filter
	Description Indicates whether the lead's <code>Title</code> field value is different from the corresponding value on its matched Data.com record (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongAddress	Type boolean
	Properties Filter, Update
	Description Indicates whether the lead's <code>Address</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongAnnualRevenue	Type boolean
	Properties Filter, Update
	Description Indicates whether the lead's <code>Annual Revenue</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongCompanyDunsNumber	Type boolean
	Properties Filter, Update
	Description Indicates whether the lead's <code>Company D-U-N-S Number</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).
IsFlaggedWrongCompanyName	Type boolean

Field Name	Details
	<p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Company Name field value is flagged as wrong to Data.com (true) or not (false).</p>
IsFlaggedWrongEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Email field value is flagged as wrong to Data.com (true) or not (false).</p>
IsFlaggedWrongIndustry	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Industry field value is flagged as wrong to Data.com (true) or not (false).</p>
IsFlaggedWrongName	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Name field value is flagged as wrong to Data.com (true) or not (false).</p>
IsFlaggedWrongNumberOfEmployees	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's No. of Employees field value is flagged as wrong to Data.com (true) or not (false).</p>
IsFlaggedWrongPhone	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	<p>Description Indicates whether the lead's <code>Phone</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsFlaggedWrongTitle</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the lead's <code>Title</code> field value is flagged as wrong to Data.com (<code>true</code>) or not (<code>false</code>).</p>
<code>IsInactive</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, <code>Filter, Group, Sort</code></p> <p>Description Indicates whether the lead has been reported to Data.com as <code>Inactive</code> (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedAddress</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the lead's <code>Address</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedAnnualRevenue</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p> <p>Description Indicates whether the lead's <code>Annual Revenue</code> field value is in a <code>Reviewed</code> state (<code>true</code>) or not (<code>false</code>).</p>
<code>IsReviewedCompanyDunsNumber</code>	<p>Type <code>boolean</code></p> <p>Properties <code>Filter, Update</code></p>

Field Name	Details
	<p>Description Indicates whether the lead's Company D-U-N-S Number field value is in a Reviewed state (true) or not (false).</p>
IsReviewedCompanyName	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Company Name field value is in a Reviewed state (true) or not (false).</p>
IsReviewedDandBCompanyDunsNumber	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's D&B Company D-U-N-S Number field value is in a Reviewed state (true) or not (false).</p>
IsReviewedEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Email field value is in a Reviewed state (true) or not (false).</p>
IsReviewedIndustry	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the lead's Industry field value is in a Reviewed state (true) or not (false).</p>
IsReviewedName	<p>Type boolean</p> <p>Properties Filter, Update</p>

Field Name	Details
	Description
	Indicates whether the lead's Name field value is in a Reviewed state (true) or not (false).
IsReviewedNumberOfEmployees	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the lead's No. of Employees field value is in a Reviewed state (true) or not (false).
IsReviewedPhone	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the lead's Phone field value is in a Reviewed state (true) or not (false).
IsReviewedTitle	Type
	boolean
	Properties
	Filter, Update
	Description
	Indicates whether the lead's Title field value is in a Reviewed state (true) or not (false).
LastMatchedDate	Type
	dateTime
	Properties
	Filter, Sort
	Description
	The date the lead record was last matched and linked to a Data.com record.
LastName	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The lead's last name.

Field Name	Details
LastStatusChangedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of who or what last changed the record's Clean_Status field value: a Salesforce user or a Clean job.</p>
LastStatusChangedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date on which the record's Clean_Status field value was last changed.</p>
Latitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with Longitude to specify the precise geolocation of a billing address. Data not currently provided.</p>
LeadId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique, system-generated ID assigned when the lead record was created.</p>
Longitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with Latitude to specify the precise geolocation of a billing address. Data not currently provided.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	Field label is Lead Clean Info Name . The name of the lead. Maximum size is 255 characters.
NumberOfEmployees	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The number of employees working at the lead.
Phone	Type
	phone
	Properties
	Filter, Group, Nillable, Sort
	Description
	The phone number for the lead.
PostalCode	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the lead.
State	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the lead.
Street	Type
	textarea
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address of the lead.
Title	Type
	string

Field Name	Details
	Properties Filter, Group, Nillable, Sort
	Description The lead's title.

Usage

Developers can create triggers that read the Lead Clean Info fields to help automate the cleaning or related processing of lead records.

LeadFeed

Represents a single feed item in the feed displayed on the detail page for a lead record. This object is available in API version 18.0 and later.

A lead feed shows recent changes to a lead record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to leads in Salesforce.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Lead object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	Type textarea Properties Nillable, Sort Description The content of LeadFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.

Field	Details
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>

Field	Details
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by CreatedDateDESC sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>

Field	Details
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default <code>community</code>, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.

Field	Details
	<ul style="list-style-type: none">• AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope.• For feed items with a record parent, users can set NetworkScope only to AllNetworks.• You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the lead record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none">• ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).

Field	Details
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none">• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RypplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• CaseCommentPost—generated event when a user adds a case comment for a case object• EmailMessageEvent—generated event when an email related to a case object is sent or received• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• ChangeStatusPost—generated event when a user changes the status of a case• AttachArticleEvent—generated event when a user attaches an article to a case <p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p>

Field	Details
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none">• AllUsers—The feed item is available to all users who have permission to see the feed item.• InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none">• For record posts, Visibility is set to InternalUsers for all internal users by default.• External users can set Visibility only to AllUsers.• On user and group posts, only internal users can set Visibility to InternalUsers.

Usage

Use this object to track changes for a lead record. If a lead record is converted, use this object to query and retrieve the associated lead feed items.

SEE ALSO:

[Lead](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

LeadHistory

Represents the history of changes to the values in the fields of a lead.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort,</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LeadId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Lead. Label is Lead ID.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to a lead.

This object respects field level security on the parent object.

SEE ALSO:

[LeadShare](#)

[LeadStatus](#)

LeadOwnerSharingRule

Represents the rules for sharing a lead with users other than the owner.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Leads owned by users in the source group trigger the rule to give access.</p>
LeadAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group. The target user or group is being given access.</p>

Usage

Use these objects to manage the sharing rules for leads. General sharing and Territory-related sharing use this object.

SEE ALSO:

[Lead](#)
[LeadShare](#)
[LeadStatus](#)
[Metadata API Developer Guide: SharingRules](#)

LeadShare

Represents a sharing entry on a Lead.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
<code>IsDeleted</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
<code>LeadAccessLevel</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the User or Group has to the Lead. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All This value is not valid when creating or updating these records.

Field	Details
	<p>This field must be set to an access level that is higher than the organization's default access level for leads.</p>
LeadId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Lead associated with this sharing entry. This field can't be updated.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write a value in this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Values include:</p> <ul style="list-style-type: none">• <code>Manual</code>—The User or Group has access because a user with “All” access manually shared the Lead with them.• <code>Owner</code>—The User is the owner of the Lead.• <code>Rule</code>—The User or Group has access via a Lead sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the Lead. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit leads owned by other users.

If you attempt to create a record that matches an existing record, the existing record is returned.

SEE ALSO:

[AccountShare](#)
[Case](#)
[CaseShare](#)
[OpportunityShare](#)

LeadStatus

Represents the status of a Lead, such as Open, Qualified, or Converted.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
IsConverted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this lead status value represents a converted lead (<code>true</code>) or not (<code>false</code>). Multiple lead status values can represent a converted lead.</p>
IsDefault	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the default lead status value (<code>true</code>) or not (<code>false</code>) in the picklist.</p>

Field	Details
MasterLabel	Type string Properties Filter, Group, Nillable, Sort Description Master label for this lead status value. This display value is the internal label that does not get translated.
SortOrder	Type int Properties Filter, Group, Nillable, Sort Description Number used to sort this value in the lead status picklist. These numbers are not guaranteed to be sequential, as some previous lead status values might have been deleted.

Usage

This object represents a value in the lead status picklist (see Lead Status Picklist). The lead status picklist provides additional information about the status of a Lead, such as whether a given status value represents a converted Lead. Query this object to retrieve the set of values in the lead status picklist, and then use that information while processing Lead objects to determine more information about a given lead. For example, the application could test whether a given lead is converted based on its `Status` value and the value of the `IsConverted` property in the associated LeadStatus record.

SEE ALSO:

[LeadOwnerSharingRule](#)
[LeadShare](#)

LeadTag

Associates a word or short phrase with a Lead.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

LeadTag stores the relationship between its parent TagDefinition and the Lead being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

LimitAllocationPerApp

Represents a connected app quota for an API limit. This object is available in API version 30.0 and later.



Note: LimitAllocationPerApp is currently available through a pilot program. For information on enabling LimitAllocationPerApp for your organization, contact Salesforce.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ConnectedAppName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the connected app.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the record in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This field is automatically generated but you can supply your own value if you create the record using the API.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>

Field Name	Details
MasterLabel	<p>Description The language of the <code>MasterLabel</code>.</p>
LimitType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
Description	<p>The type of API limit for which you're setting a quota. The available values are:</p> <ul style="list-style-type: none"> • <code>TotalRequests</code>—Total API requests per 24-hour period • <code>ApiBatchItems</code>—Number of Bulk API batches per 24-hour period • <code>StreamEventsPerDay</code>—Number of Streaming API events per 24-hour period • <code>GenStreamingEventsPerDay</code>—Number of generic streaming events per 24-hour period
	<p>Note: Generic streaming is currently available through a pilot program. For information on enabling generic streaming for your organization, contact Salesforce.</p>
MasterLabel	<p>Type string</p>
Properties	<p>Create, Filter, Group, Sort, Update</p>
Description	<p>Master label for the API limit quota. This is the internal label that does not get translated.</p>
Percentage	<p>Type int</p>
Properties	<p>Create, Filter, Group, Sort, Update</p>
Description	<p>The percentage of the API limit reserved for the connected app. The app can't consume more than this percentage of the API limit.</p>
You can find the available percentage for each limit type through the user interface or by using the <code>query()</code> call.	

Usage

App quotas for API limits enable you to reserve API capacity for mission-critical connected apps or set a ceiling for API usage of non-critical connected apps.

LineitemOverride

A forecast override of a line item on an Opportunity. This read-only object for customizable forecasting has a child-parent relationship with OpportunityOverride.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

This object is only accessible if your organization has enabled the customizable forecasting feature, which can be done in the user interface. Requires the "View All Data" permission.

Fields

Field	Details
AmountInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden amount rolls up through the forecast hierarchy (<code>true</code>), or was overridden by the owner of this record (<code>false</code>).</p>
ForecastCategoryInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden forecast category rolls up through the forecast hierarchy (<code>true</code>), or was overridden by the owner of this record (<code>false</code>).</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>

Field	Details
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the associated Opportunity.</p>
OpportunityLineItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the associated OpportunityLineItem.</p>
OverrideAmount	<p>Type currency</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The total monetary amount of the line item, which may be overridden.</p>
OverrideForecastCategory	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The forecast category of the line item. Can be overridden.</p>
OverrideQuantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The quantity of the line item, which may be overridden.</p>
OverrideUnitPrice	<p>Type currency</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The monetary amount of the unit price. Can be overridden.</p>

Field	Details
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the owner of this record.</p>
QuantityInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden quantity rolls up through the forecast hierarchy (<code>true</code>), or is overridden by the owner of this record (<code>false</code>).</p>
UnitPriceInherited	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the overridden unit price rolls up through the forecast hierarchy (<code>true</code>), or was overridden by the owner of this record (<code>false</code>).</p>

SEE ALSO:

[OpportunityOverride](#)

LinkedArticle

Represents a Knowledge article that is attached to a record. This object is available in API version 37.0 and later.

As of Summer '16, linked articles are supported only for work orders and work order line items.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(),
query(), retrieve(), undelete(), update(), upsert()
```

Special Access Rules

Knowledge must be set up in your org.

Fields

Field Name	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the Knowledge article attached to the record. The label in the user interface is Knowledge Article ID.</p>
KnowledgeArticleVersionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The version of the Knowledge article attached to the record. This field lists the title of the attached version and links to the version. The label in the user interface is Article Version. When you attach an article to a work order, that version of the article stays associated with the work order, even if later versions are published. If needed, you can detach and reattach an article to a work order to link the latest version.</p>
LinkedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the record that the Knowledge article is attached to. The label in the user interface is Linked Record ID.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description The title of the article. The label in the user interface is Article Title.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the article's record type, if used.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The type of record that the Knowledge article is attached to. For example, work order. The label in the user interface is Linked Object Type.</p>

Usage

Admins can customize linked articles' page layouts, fields, validation rules, and more from the Linked Articles node in Setup under Knowledge.

LinkedArticleFeed

Represents a single feed item on a linked Knowledge article attached to a record. This object is available in API version 37.0 and later. As of Summer '16, linked articles are supported only for work orders and work order line items.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Knowledge must be set up in your org.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the feed item. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with the feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags.</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Although the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><i></code> • <code><u></code>

Field Name	Details
	<ul style="list-style-type: none"> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p>
LikeCount	Type int
	Properties Filter, Group, Sort
	Description The number of Likes associated with the feed item.
LinkUrl	Type url
	Properties Nillable, Sort
	Description The URL of a LinkPost .
NetworkScope	Type picklist
	Properties Group, Nillable, Restricted picklist, Sort
	Description Specifies whether the feed item is available in the default community , a specific community, or all communities.
ParentId	Type reference
	Properties Filter, Group, Sort
	Description The ID of the object type to which the feed item is related. For example, set this field to a UserId to post to someone's profile feed. Or set it to an AccountId to post to a specific account.

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description The ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or only internal users.</p>

LinkedArticleHistory

Represents the history of changes made to tracked fields on a linked article. This object is available in API version 37.0 and later. As of Summer '16, linked articles are supported only for work orders and work order line items.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Knowledge must be set up in your org.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
LinkedArticleId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the tracked linked article. The history is displayed on the detail page for this record.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>

ListView

Represents a list view. A list view specifies a set of records for an object, based on specific criteria. This object is available in API version 32.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Name	Details
CreatedBy	<p>Type User</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The name of the user who created the list view.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The fully qualified developer name of the list view.</p>
IsSoqlCompatible	<p>Type boolean</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description Whether the list view can be used with SOQL..</p>
LastModifiedBy	<p>Type User</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The name of the user who last modified the list view.</p>
LastModifiedById	<p>Type ID</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The ID of the user who last modified the list view.</p>

Name	Details
LastModifiedDate	<p>Type dateTime</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The date and time of the last modification of the list view, with a precision of one second.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The date and time when the list view was last referenced, with a precision of one second.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The date and time when the list view was last viewed, with a precision of one second.</p>
Name	<p>Type string</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The name of the list view.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The namespace of the list view.</p>
sObjectType	<p>Type string</p> <p>Properties Filter, Query, Retrieve, Sort</p> <p>Description The API name of the sObject for the list view.</p>

ListViewChart

Represents a graphical chart that's displayed on Salesforce1 list views. The chart aggregates data that is filtered based on the list view that's currently displayed. This object is available in API version 33.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Name	Description
AggregateField	<p>Type string</p> <p>Properties Create, Filter, Query, Restricted picklist, Retrieve, Sort, Update</p> <p>Description The field that's used for calculating data on each group. <code>AggregateField</code> can't be the same as <code>GroupingField</code>.</p>
AggregateType	<p>Type picklist</p> <p>Properties Create, Filter, Query, Retrieve, Sort, Update</p> <p>Description The type of calculations to run on each group. The supported <code>AggregateType</code> values are <code>Count</code>, <code>Sum</code>, and <code>Avg</code>.</p>
ChartType	<p>Type picklist</p> <p>Properties Create, Filter, Query, Retrieve, Sort, Update</p> <p>Description The type of chart to create. The supported chart types are horizontal bar chart, vertical bar chart, and donut chart.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Query, Retrieve, Sort, Update</p> <p>Description The fully qualified developer name of the chart.</p>

Name	Description
GroupingField	<p>Type string</p> <p>Properties Create, Filter, Query, Retrieve, Sort, Update</p> <p>Description The field that's used to divide the data into collections. The field has to be supported by SOQL GROUP BY functionality. GroupingField can't be the same as AggregateField.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Query, Retrieve, Sort, Update</p> <p>Description The label for the chart.</p>
OwnerId	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Query, Retrieve, Sort</p> <p>Description The ID of the user who owns the chart.</p>
sObjectType	<p>Type string</p> <p>Properties Create, Filter, Query, Retrieve, Sort</p> <p>Description The API name of the sObject for the chart.</p>

ListViewChartInstance

Retrieves metadata for all standard and custom charts for a given entity in context of a given list view. This object is available in API versions 34.0 and later.

Supported Calls

`describeSObjects(), query()`

Fields

Field Name	Details
AggregateField	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The field that's used for calculating data on each group. <code>AggregateField</code> can't be the same as <code>GroupingField</code>.</p>
AggregateType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of calculations to run on each group. The supported <code>AggregateType</code> values are <code>Count</code>, <code>Sum</code>, and <code>Avg</code>.</p>
ChartType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of chart to create. The supported chart types are horizontal bar chart, vertical bar chart, and donut chart.</p>
DataQuery	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The SOQL query that can be executed to fetch the data for drawing a chart.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description API name of the chart. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package</p>

Field Name	Details
	<p>installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
ExternalId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Reserved for future use.</p>
GroupingField	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The field that's used to divide the data into collections. The field has to be supported by SOQL GROUP BY functionality. GroupingField can't be the same as AggregateField.</p>
IsDeletable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates if the chart can be deleted.</p>
IsEditable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates if the chart can be edited. Standard charts are not editable.</p>
IsLastViewed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates if a chart is the last viewed by a user.</p>

Field Name	Details
Label	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The display name of the chart.</p>
ListViewChartId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the chart created by a user. For standard charts, this is null.</p>
ListViewContextId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the list view in context of which the chart is generated. Required to query ListViewChartInstance.</p>
SourceEntity	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description API name of the entity to which the chart is related. Required to query ListViewChartInstance.</p>

Usage

Example 1. Retrieve all custom and standard charts for Account entity for All Accounts list view

```
SELECT AggregateField, AggregateType, ChartType, DataQuery, DeveloperName, ExternalId,
GroupingField, Id, IsDeletable, IsEditable, IsLastViewed, Label, ListViewChartId,
ListViewContextId, SourceEntity FROM ListViewChartInstance WHERE SourceEntity='Account'
and ListViewContextId='00BR000000U8Hr'
```

Example 2. Retrieve metadata for a specific custom chart by ID for Account entity and All Accounts list view

```
SELECT AggregateField, AggregateType, ChartType, DataQuery, DeveloperName, ExternalId,
GroupingField, Id, IsDeletable, IsEditable, IsLastViewed, Label, ListViewChartId,
```

```
ListViewContextId, SourceEntity FROM ListViewChartInstance WHERE SourceEntity='Account'
and ListViewContextID='00BR0000000U8Hr' and ListViewChartId='0DdR0000004CBxKAM'
```

Example 3. Retrieve metadata for a specific standard chart by its developer name for Account entity and All Accounts list view

```
SELECT AggregateField, AggregateType, ChartType, DataQuery, DeveloperName, ExternalId,
GroupingField, Id, IsDeletable, IsEditable, IsLastViewed, Label, ListViewChartId,
ListViewContextId, SourceEntity FROM ListViewChartInstance WHERE SourceEntity='Account'
and ListViewContextID='00BR0000000U8Hr' and DeveloperName='AccountsByIndustry'
```

LiveAgentSession

This object is automatically created for each Live Agent session and stores information about the session. This object is available in API versions 28.0 and later.



Note: Standard fields for the LiveAgentSession object can only be modified if your administrator has given you editing permissions for these records.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update()
```

Fields

Field Name	Details
AgentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the agent associated with the session.</p>
ChatReqAssigned	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The number of chat requests that were assigned to an agent during a session.</p>
ChatReqDeclined	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The number of chat requests that were declined by an agent during a session.</p>
ChatReqEngaged	<p>Type int</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The number of chats in which an agent was engaged during a session.</p>
ChatReqTimedOut	<p>Type int</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The number of chat requests that timed out in an agent's queue during a session.</p>
LastReferencedDate	<p>Type dateTime</p>
	<p>Properties Filter, Nillable, Sort</p>
	<p>Description The date and time that the session record was last referenced.</p>
LastViewedDate	<p>Type dateTime</p>
	<p>Properties Filter, Nillable, Sort</p>
	<p>Description The date and time that the session record was last viewed.</p>
LoginTime	<p>Type dateTime</p>
	<p>Properties Create, Filter, Sort, Update</p>
	<p>Description The date and time an agent logged in during the session.</p>
LogoutTime	<p>Type dateTime</p>
	<p>Properties Create, Filter, Sort, Update</p>

Field Name	Details
	Description
	The date and time an agent logged out during a session.
Name	Type
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookupSort
	Description
	The name of the session.
NumFlagLoweredAgent	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The number of assistance flags lowered by the agent.
NumFlagLoweredSupervisor	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The number of assistance flags lowered by the supervisor.
NumFlagRaised	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The number of assistance flags raised by the agent.
OwnerId	Type
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	ID of the owner of the session record.
TimeAtCapacity	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description
	<p>The amount of time an agent spent with the maximum number of chats in his or her queue.</p>
TimeIdle	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The amount of time an agent spent idle during the session.</p>
TimeInAwayStatus	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The amount of time an agent spent with a status of "Away" during a session.</p>
TimeInChats	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The amount of time an agent spent engaged in chats during a session.</p>
TimeInOnlineStatus	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The amount of time an agent spent with a status of "Online" during a session.</p>

Usage

Use this object to query and manage live chat session records.

LiveAgentSessionHistory

This object is automatically created for each Live Agent session and stores information about changes made to the session. This object is available in API versions 28.0 and later.



Note: Standard fields for the LiveAgentSession object can only be modified if your administrator has given you editing permissions for these records.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed in a session record.</p>
LiveAgentSessionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the session record that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The original value of the field that was changed.</p>

Usage

Use this object to identify changes to live chat session records.

LiveAgentSessionOwnerSharingRule

Represents the rules for sharing a Live Agent session record with users other than the record owner. This object is available in API version 28.0 and later.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, restrictedPicklist, Sort, Update</p> <p>Description Determines the level of access users have to session records. Specifies whether or not users can read, edit, or transfer session records. Corresponds to the Default Access column in the UI.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the UI.</p>

Field	Details	
		<p>Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	Type reference	
Properties		
Create, Filter, Group, Sort		
Description		
The ID representing the source group. Session records owned by users in the source group trigger the rule to give access.		
Name	Type string	
Properties		
Create, Filter, Group, idLookup, Sort, Update		
Description		
Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the UI.		
UserOrGroupId	Type reference	
Properties		
Create, Filter, Group, Sort		
Description		
The ID representing the target user or group. The target user or group is being given access.		

Usage

Use this object to manage the sharing rules for Live Agent session records.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

LiveAgentSessionShare

This object is automatically created for each Live Agent session and stores information about the session. This object is available in API versions 28.0 and later.

 **Note:** Standard fields for the LiveAgentSession object can only be modified if your administrator has given you editing permissions for these records.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the LiveAgentSession. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All (This value is not valid for <code>create()</code> or <code>update()</code> calls.) <p>This value must be set to an access level that is higher than the organization's default access level for live chat transcripts.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the parent object, if any.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Values can include:</p> <ul style="list-style-type: none"> • Manual—The user or group has access because a user with "All" access manually shared the LiveAgentSession with them. • Owner—The user is the owner of the LiveAgentSession or is in a role above the QuickText owner in the role hierarchy.

Field Name	Details
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that has been given access to the LiveAgentSession.</p>

Usage

This object lets you determine which users and groups can view and edit LiveAgentSession records owned by other users.

If you attempt to create a new record that matches an existing record, the `create()` call updates any modified fields and returns the existing record.

LiveChatBlockingRule

Represents a rule for blocking chat visitors' IP addresses from starting new chats with agents. This object is available in API version 34.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
Description	<p>Type string</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the blocking rule—for example, the reason why the given IP address or range of addresses is being banned from starting new chats.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with</p>

Field Name	Details
	<p>a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
<code>FromIpAddress</code>	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The IP address of the user that you want to block, or the beginning of the range of IP addresses you want to block. If you want to block a range of IP addresses, indicate the end of the range in the <code>ToIpAddress</code> field. If you don't indicate an IP address in the <code>ToIpAddress</code> field, the only IP address that will be blocked is the IP address in the <code>FromIpAddress</code> field.</p>
<code>Language</code>	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the blocking rule.</p>
<code>MasterLabel</code>	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label for the blocking rule.</p>
<code>ToIpAddress</code>	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description (Optional) The end of the range of IP addresses that you want to block. The range begins with and includes the IP address in the <code>FromIpAddress</code> field, and it ends with and includes the IP address in the <code>ToIpAddress</code> field.</p>

Usage

Use this object to query and manage rules for blocking customers from starting new chats with agents.

LiveChatButton

Represents a button that allows visitors to request chats with Live Agent users. This object is available in API version 24.0 and later.

Supported Calls

`create()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
Animation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The type of animation used when an automated chat invitation appears on-screen. For automated chat invitations only. Available in API version 29.0 and later.</p>
AutoGreeting	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The text that is automatically sent from an agent to a visitor when a chat session starts.</p> <p> Note: A greeting message in the <code>AutoGreeting</code> field of the <code>LiveChatButton</code> object overrides individual users' greeting messages in the <code>AutoGreeting</code> field in the <code>LiveChatUserConfig</code> object.</p>
ChatPageId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record ID of the custom Visualforce page that contains the custom chat window code.</p>

Field Name	Details
CustomAgentName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The custom name of the agent associated with the button. Available in API version 29.0 and later.</p> <p> Note: A custom agent name in the <code>CustomAgentName</code> field of the <code>LiveChatButton</code> object overrides individual users' custom agent name in the <code>CustomAgentName</code> field in the <code>LiveChatUserConfig</code> object.</p>
CustomRoutingClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the Apex class you customize to create your custom routing rules if <code>RoutingType</code> equals <code>Custom</code>. Available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
HasQueue	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether to allow incoming chat requests to queue until an agent is available.</p>

Field Name	Details
InviteEndPosition	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort, Update</p> <p>Description The position on-screen where an automated chat invitation's animation ends.</p> <p> Note: You don't need to select an end position for your automated chat invitation if you use a custom animation.</p> <p>For automated chat invitations only. Available in API version 29.0 and later.</p>
InviteImageId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The record ID of the static image resource displayed on your automated chat invitation. For automated chat invitations only. Available in API version 29.0 and later.</p>
InviteStartPosition	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort, Update</p> <p>Description The position on-screen where an automated chat invitation's animation begins.</p> <p> Note: You don't need to select a start position for your automated chat invitation if you use a custom animation.</p> <p>For automated chat invitations only. Available in API version 29.0 and later.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description For automated chat invitations, specifies whether an automated chat invitation is active and able to be sent to customers (<code>true</code>) or not (<code>false</code>). For chat buttons, this is set to <code>true</code> by default.</p>
Language	<p>Type picklist</p>

Field Name	Details
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	The language of the chat.
MasterLabel	Type
	string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Label for the live chat button.
NumberOfReroutingAttempts	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Specifies the number of times a chat request can be rerouted to available agents if all agents reject the chat request.
OfflineImageId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The record ID of the static image resource that is displayed when the button is offline (inactive).
OnlineImageId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The record ID of the static image resource that is displayed when the button is online (active).
OptionsHasInviteAfterAccept	Type
	boolean
	Properties
	Create, Filter, Update

Field Name	Details
	<p>Description</p> <p>Specifies whether an automated chat invitation can be sent to a customer after that customer has accepted a prior automated chat invitation (<code>true</code>) or not (<code>false</code>). For automated chat invitations only. Available in API version 29.0 and later.</p>
OptionsHasInviteAfterReject	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Specifies whether an automated chat invitation can be sent to a customer after that customer has rejected a prior automated chat invitation (<code>true</code>) or not (<code>false</code>). For automated chat invitations only. Available in API version 29.0 and later.</p>
OptionsHasRerouteDeclinedRequest	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Specifies whether a chat request that has been rejected by all available agents should be rerouted to available agents again (<code>true</code>) or not (<code>false</code>).</p>
OptionsIsAutoAccept	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Specifies whether a chat request should be automatically accepted by the agent it's assigned to (<code>true</code>) or not (<code>false</code>). For chat buttons and automated chat invitations with <code>RoutingType</code> set to <code>Most Available</code> or <code>Least Active</code>. Available in API version 30.0 and later.</p>
OptionsIsInviteAutoRemove	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Specifies whether an automated chat invitation should be automatically removed from the screen after a certain amount of time (<code>true</code>) or not (<code>false</code>). For automated chat invitations only. Available in API version 29.0 and later.</p>

Field Name	Details
OverallQueueLength	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The maximum number of chat requests allowed to queue.</p>
PerAgentQueueLength	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The maximum number of chat requests allowed to queue for each agent with the required skill.</p>
PostchatPageId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record ID of the custom Visualforce page displayed when the chat ends.</p>
PostchatUrl	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The URL the user is directed to after the chat ends.</p>
PrechatFormPageId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record ID of the custom Visualforce page displayed before the chat begins.</p>
PrechatFormUrl	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	Description
	<p>The URL the user is directed to before the chat begins.</p>
PushTimeout	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The number of seconds an agent has to answer a chat request before it's routed to the next available agent.</p>
RoutingType	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	<p>How chat requests are routed to agents. The values are:</p>
	<ul style="list-style-type: none"> • Choice—Incoming chat requests are added to the queue in Live Agent in the Salesforce console and are available to any agent with the required skill. • Least Active—Incoming chats are routed to the agent with the required skill who has the fewest active chats. • Most Available—Incoming chats are routed to the agent with the required skill and the greatest difference between chat capacity and active chat sessions. For example, if Agent A and Agent B each have a chat capacity of five, and Agent A has three active chat sessions while Agent B has one, incoming chats will be routed to Agent B. • Custom—Incoming chats are routed to agents based on the custom routing rules specified in the Apex class you use to create them.
SiteId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	<p>The record ID of the site used for loading static resources and custom Visualforce pages.</p>
SkillId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	<p>Description The record ID of the skill used to route incoming chat requests. To associate multiple skills with a live chat button, reference one skill in the <code>SkillId</code> field and use <code>LiveChatButtonSkill</code> junction objects for the remaining skills.</p>
TimeToRemoveInvite	<p>Type int</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The number of seconds an automated invitation stays on-screen before it is automatically removed. For automated chat invitations only. Available in API version 29.0 and later.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of element to display to customers (either a chat button or an automated invitation).</p>
WindowLanguage	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The language used for standard chat windows. Custom chat windows use the language of the user's browser.</p>

Usage

Use this object to query and manage live chat buttons and automated chat invitations.

LiveChatButtonDeployment

Associates a Live Agent automated chat invitation with a specific deployment. This object is available in API versions 28.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
ButtonId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the automated invitation associated with the deployment.</p>
DeploymentId	<p>Type reference</p> <p>Properties Create, Filter, Group, NillableSort</p> <p>Description The ID of the deployment that will feature the automated invitation.</p>

Usage

Use this object to associate automated chat invitations with specific deployments.

LiveChatButtonSkill

Represents all the skills available to a LiveChatButton except the one currently assigned. To retrieve the skill currently assigned, query LiveChatButton. This object is available in API version 25.0 and later.

Supported Calls

`create(), delete(), query(), update()`

Fields

Field Name	Details
ButtonID	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The record ID of the button.</p>

Field Name	Details
SkillID	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description The record ID of the skill.</p>

Usage

Use this object to assign a specific skill to a specific button for multi-skill routing. For example:

```
String myButtonId = "button_Id";
String myButtonDevName = "button_DeveloperName";
List<String> skillIds = new List<String>();

//Get one skill ID from button
for(LiveChatButton lcb : [SELECT SkillId FROM LiveChatButton WHERE DeveloperName =: myButtonDevName]) {
    skillIds.add(lcb.SkillId);
}

//Get remaining skills from LiveChatButtonSkill join object
for(LiveChatButtonSkill lcbs : [SELECT SkillID FROM LiveChatButtonSkill WHERE ButtonId =: myButtonId]) {
    skillIds.add(lcbs.SkillId);
}

//Retrieve all skills into a single list
List<Skill> skills = [SELECT Id, DeveloperName FROM Skill WHERE Id IN :skillIds];
```

LiveChatDeployment

Represents the general settings for deploying Live Agent on a website. This object is available in API version 24.0 and later.

Supported Calls

`create()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
BrandingId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p>

Field Name	Details
ConnectionTimeoutDuration	Description The record ID of the static image resource that's displayed in the chat window
ConnectionWarningDuration	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description Indicates the amount of time before the chat times out, in seconds.
DeveloperName	Type string Properties Create, Filter, Group, Sort, Update Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.  Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.
Domains	Type textarea Properties Create, Filter (unavailable in API version 25.0 and later), Nillable, Sort (unavailable in API version 25.0 and later), Update Description A comma-separated list of domains the deployment is whitelisted for. Leave this blank to allow the deployment to be used on any domain.

Field Name	Details
HasTranscriptSave	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether visitors can download and save transcripts from the chat window</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the deployment</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the deployment</p>
MobileBrandingId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record ID of the static image resource displayed in the mobile version of the chat window</p>
OptionsHasPrechatApi	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether developers can access the Pre-Chat API</p>
SiteId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The record ID of the site used for loading static resources</p>
WindowTitle	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text displayed in the title bar of the browser window used to launch the chat window</p>

Usage

Use this object to query and manage live chat deployments.

LiveChatSensitiveDataRule

Represents a rule for masking or deleting data of a specified pattern. Written as a regular expression (regex). This object is available in API version 35.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
ActionType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklistSort, Update</p> <p>Description The action to take on the text (remove or replace) when the sensitive data rule is triggered.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p>

Field Name	Details
	Description
	The description of the sensitive data rule—for example, “Block social security numbers.”
DeveloperName	Type string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object’s name in a managed package and the changes are reflected in a subscriber’s organization.
	 Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.
EnforceOn	Type int
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Determines the roles on which the rule is enforced. The value is determined using bitwise OR operation. There are seven possible values:
	<ol style="list-style-type: none"> 1. Rule enforced on Agent 2. Rule enforced on Visitor 3. Rule enforced on Agent and Visitor 4. Rule enforced on Supervisor 5. Rule enforced on Agent and Supervisor 6. Rule enforced on Visitor and Supervisor 7. Rule enforced on Agent, Visitor, and Supervisor
IsEnabled	Type boolean
	Properties
	Create, Filter, Group, Sort, Update

Field Name	Details
	Description
	<p>Specifies whether a sensitive data rule is active (<code>true</code>) or not (<code>false</code>). Default value (if none is provided) is <code>false</code>.</p>
Language	Type picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The language of the sensitive data rule.
MasterLabel	Type string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Label for the sensitive data rule.
NamespacePrefix	Type string
	Properties
	Filter, Nillable
	Description
	<p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p>
	The namespace prefix can have one of the following values:
	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Pattern	Type textarea
	Properties
	Create, Update

Field Name	Details
	<p>Description The pattern of text blocked by the rule. Written as a JavaScript regular expression (regex).</p>
Replacement	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The string of characters that replaces the blocked text (if <code>ActionType Replace</code> is selected).</p>

Usage

Use this object to mask or delete data of specified patterns, such as credit card, social security, phone and account numbers, or even profanity.

LiveChatTranscript

This object is automatically created for each Live Agent chat session and stores information about the session. This object is available in API version 24.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
Abandoned	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time in seconds an incoming chat request remained unanswered by an agent before the chat was disconnected by the customer.</p>
AccountId	<p>Type ID</p>

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description ID of the account associated with the chat transcript.
AverageResponseTimeOperator	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The agent's average response time (in seconds) to chat messages from the visitor.
AverageResponseTimeVisitor	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The visitor's average response time (in seconds) to chat messages from the agent.
Body	Type textarea
	Properties Create, Nillable, Update
	Description The contents of the chat.
Browser	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The browser the visitor used for the chat.
BrowserLanguage	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The language of the visitor's browser.
CaseID	Type reference

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The ID of the case associated with the chat transcript.</p>
ChatDuration	<p>Type int</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The total duration of the chat in seconds.</p>
ChatKey	<p>Type string</p>
	<p>Properties Create, Filter, Group, idLookup, Nillable, Sort</p>
	<p>Description The session ID of the chat before it is persisted. ChatKey can be used with advanced integrations in the Salesforce console. This field is available in API version 25.0 and later.</p>
ContactID	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The ID of the contact associated with the chat transcript.</p>
EndedBy	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>
	<p>Description The way the chat was ended: by the operator, the visitor, or the system.</p>
EndTime	<p>Type dateTime</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description The time the chat ended.</p>

Field Name	Details
IpAddress	Type string
	Properties Create, Filter, Nillable, Sort, Update
	Description The visitor's IP address.
LastReferencedDate	Type date
	Properties Filter, Nillable, Sort, Update
	Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date
	Properties Filter, Nillable, Sort, Update
	Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
LeadID	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ID of the lead associated with the chat transcript.
LiveChatButtonID	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ID of the LiveChatButton the chat session originated from.
LiveChatDeploymentID	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Type
	The ID of the LiveChatDeployment the chat session originated from.
LiveChatVisitorID	Type
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	The ID of the visitor associated with the chat transcript.
Location	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The best-guess approximation of the visitor's location.
MaxResponseTimeOperator	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The maximum time in seconds it took an agent to respond to a chat visitor's message.
MaxResponseTimeVisitor	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	The maximum time in seconds it took a customer to respond to an agent's message.
Name	Type
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description
	The name of the transcript.
OperatorMessageCount	Type
	int

Field Name	Details
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description The number of messages sent by agent(s) during the chat.
OwnerID	Type reference
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description The ID of the operator who participated in the chat last; for missed chats, this is a system user.
Platform	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description The visitor's operating system platform.
ReferrerUri	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description Where the chat request originated.
RequestTime	Type dateTime
	Properties Create, Filter, Nullable, Sort, Update
	Description The time the visitor requested the chat.
ScreenResolution	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description The visitor's screen resolution.

Field Name	Details
SkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The record ID of the primary Skill associated with the LiveChatButton the chat session originated from. To associate multiple skills with a LiveChatTranscript, use LiveChatTranscriptSkill junction objects.</p>
StartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The time the chat started.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The final status of the chat: completed, missed, dropped or blocked.</p>
SupervisorTranscriptBody	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The text body of the supervisor's chat transcript.</p>
UserAgent	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The visitor's user agent string.</p>
VisitorMessageCount	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The number of messages sent by the visitor during the chat.</p>
VisitorNetwork	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The network or service provider the chat visitor used for the chat.</p>
WaitTime	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The total amount of time in seconds a chat request was waiting to be accepted by an agent.</p>

Usage

Use this object to query and manage live chat transcripts.

LiveChatTranscriptEvent

Captures specific events that occur over the lifetime of a chat. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
AgentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the agent associated with the event.</p>

Field Name	Details
Detail	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Details associated with the event.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
LiveChatTranscriptId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the live chat transcript associated with the event.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the event.</p>
Time	<p>Type dateTime</p> <p>Properties Create, Filter, Sort</p>

Field Name	Details
	<p>Description The time at which the event happened.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The kind of event that occurred.</p>

Usage

Use this object to query and manage live chat transcript events.

LiveChatTranscriptHistory

Represents changes to field values on a LiveChatTranscript object. This object is available in API version 24.0 and later.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed</p>
LiveChatTranscriptID	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the transcript that was changed</p>

Field Name	Details
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The old value of the field that was changed</p>

Usage

Use this object to identify changes to a live chat transcript.

LiveChatTranscriptOwnerSharingRule

Represents the rules for sharing a Live Agent chat transcript record with users other than the record owner. This object is available in API version 29.0 and later.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), getDeleted(), getUpdated(), query(), retrieve(), update()`

Fields

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, restrictedPicklist, Sort, Update</p>

Field	Details
	<p>Description Determines the level of access users have to chat transcript records. Specifies whether or not users can read, edit, or transfer chat transcript records. Corresponds to the Default Access column in the UI.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the UI.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Chat transcript records owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the UI.</p>

Field	Details
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group. The target user or group is being given access.</p>

Usage

Use this object to manage the sharing rules for Live Agent chat transcript records.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

LiveChatTranscriptShare

Represents a sharing entry on a LiveChatTranscript object. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the LiveChatTranscript. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All (This value is not valid for <code>create()</code> or <code>update()</code> calls.) <p>This value must be set to an access level that is higher than the organization's default access level for live chat transcripts.</p>

Field Name	Details
ParentId	Type reference
	Properties Create, Filter, Group, Sort Description ID of the parent object, if any
RowCause	Type picklist
	Properties Filter, Group, Restricted picklist, Sort Description Reason that this sharing entry exists.
	You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.
	Values may include:
	<ul style="list-style-type: none"> • Manual—The User or Group has access because a user with “All” access manually shared the LiveChatTranscript with them. • Owner—The User is the owner of the LiveChatTranscript or is in a role above the QuickText owner in the role hierarchy.
UserOrGroupID	Type reference
	Properties Create, Filter, Group, Sort Description ID of the User or Group that has been given access to the LiveChatTranscript.

Usage

This object lets you determine which users and groups can view and edit LiveChatTranscript records owned by other users.

If you attempt to create a new record that matches an existing record, the `create()` call updates any modified fields and returns the existing record.

LiveChatTranscriptSkill

Represents a join between LiveChatTranscript and Skill. This object is available in API version 25.0 and later.

Supported Calls

`create(), delete(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update()`

Fields

Field Name	Details
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The name of the transcript.
SkillId	Type reference Properties Create, Filter, Group, Nullable, Sort Description The record ID of the skill.
TranscriptId	Type reference Properties Create, Filter, Group, Sort Description The record ID of the transcript.

Usage

Use this object to assign a specific skill to a specific transcript for multi-skill routing.

LiveChatUserConfig

Represents a setting that controls the console settings for Live Agent users. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
AutoGreeting	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The text that is automatically sent from an agent to a visitor when a chat session starts.</p>
Capacity	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Limits the number of active chat session an agent can engage in.</p>
CriticalWaitTime	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The amount of time before a chat flashes to alert an agent to answer it.</p>
CustomAgentName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The custom name of the agent associated with the Live Agent configuration.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the</p>

Field Name	Details
	<p>object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
HasCrmChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the CRM chatlet appears in the agent console.</p>
HasDetailsChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the Details chatlet appears in the agent console.</p>
HasHistoryChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the History chatlet appears in the agent console.</p>
HasKnowledgeChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the Knowledge chatlet appears in the agent console.</p>
HasLivetrackChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the Livetrack chatlet appears in the agent console.</p>

Field Name	Details
HasLogoutSound	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether a sound plays when an agent logs out of the console.</p>
HasMapChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the Map chatlet appears in the agent console.</p>
HasNotifications	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether desktop notifications are enabled for the configuration. Available in API version 25.0 and later.</p>
HasQueueChatlet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether the Queue chatlet appears in the agent console.</p>
HasRequestSound	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Determines whether a sound plays when a chat request comes in.</p>
HasSneakPeek	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	<p>Determines whether an agent sees a real-time preview of the messages a visitor types.</p>
IsAutoAwayOnDecline	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	<p>Determines whether agents' status is automatically changed to Away when they decline a chat request. Available in API version 26.0 and later.</p>
Language	Type
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The language of the configuration.
MasterLabel	Type
	string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	The name of the configuration.
OptionsHasAgentFileTransfer	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	<p>Determines whether agents can initiate a file transfer from a chat customer. Available in API version 31.0 and later.</p>
OptionsHasAgentSneakPeek	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	<p>Determines whether Sneak Peek is enabled for agents. Available in API version 29.0 and later.</p>

Field Name	Details
OptionsHasChatConferencing	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether agents can invite other agents into a customer chat. Available in API version 34.0 and later.</p>
OptionsHasChatMonitoring	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether supervisors can view agents' ongoing chats. Available in API version 29.0 and later.</p>
OptionsHasChatTransferToAgent	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether an agent can transfer a chat directly to another agent. Available in API version 36.0 and later.</p>
OptionsHasChatTransferToButton	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether an agent can transfer a chat to an agent assigned to a particular chat button. Available in API version 36.0 and later.</p>
OptionsHasChatTransferToSkill	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether an agent can transfer a chat to agents assigned to a particular skill. Available in API version 36.0 and later.</p>
OptionsHasVisitorBlocking	<p>Type boolean</p>

Field Name	Details
	Properties Create, Filter, Update
	Description Determines whether an agent has the ability to block troublesome visitors by IP address. Available in API version 34.0 and later.
OptionsHasWhisperMessage	Type boolean
	Properties Create, Filter, Update
	Description Determines whether supervisors can send private messages to agents within an agent's chat with a customer. Available in API version 29.0 and later.
OptionsIsAutoAwayOnPushTimeout	Type boolean
	Properties Create, Filter, Update
	Description Determines whether an agent's status automatically changes to Away if the agent doesn't respond to a chat request within the specified push time-out limit. Available in API version 34.0 and later.
SupervisorDefaultAgentStatus	Type picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The default agent status by which to filter agents in the Agent Status list in the supervisor panel.
SupervisorDefaultButtonId	Type picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The default button ID by which to filter agents in the Agent Status list in the supervisor panel.
SupervisorDefaultSkillId	Type picklist

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>
	<p>Description The default skill ID by which to filter agents in the Agent Status list in the supervisor panel.</p>

Usage

Use this object to query and manage agent configurations in Live Agent.

LiveChatUserConfigProfile

Represents a join between LiveChatUserConfig and Profile. This object is available in API version 24.0 and later.

Supported Calls

`create() delete() query() retrieve() update() upsert()`

Fields

Field Name	Details
LiveChatUserConfigId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The record ID of the agent configuration</p>
ProfileId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The record ID of the profile</p>

Usage

Use this object to assign specific agent configurations to specific user profiles.

LiveChatUserConfigUser

Represents a join between LiveChatUserConfig and User. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
LiveChatUserConfigId	Type reference Properties Create, Filter, Group, Sort Description The record ID of the agent configuration
UserId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The record ID of the user

Usage

Use this object to assign specific agent configurations to specific users.

LiveChatVisitor

Represents a website visitor who has started or tried to start a chat session. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The name of the visitor</p>
SessionKey	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The session key used to uniquely identify the visitor</p>

Usage

Use this object to query and manage live chat visitors.

LoginEvent

Represents a trackable user login event in your org. This object is available in API version 36.0 and later.

Supported Calls

query()

Special Access Rules

Accessing this object requires View Login Forensics Events and API Enabled user permissions.

Fields

Field	Details
AdditionalInfo	<p>Type string</p> <p>Properties Nillable</p> <p>Description JSON serialization of additional information that's captured from the <code>HTTP</code> headers during a login request. For example, <code>{"field1": "value1", "field2": "value2"}</code>. See Working with AdditionalInfo on page 1205.</p>
ApiType	<p>Type string</p> <p>Properties Nillable</p> <p>Description English-only. The type of API that's used to log in. Values include:</p> <ul style="list-style-type: none">• XML-RPC• OSOAP• SOAP Enterprise• SOAP Partner• SOAP Internal/CrossInstance• REST API• Metadata API• N/A
ApiVersion	<p>Type string</p> <p>Properties Nillable</p> <p>Description English-only. The version number of the API. If no version number is available, "Unknown" is returned.</p>

Field	Details
Application	<p>Type string</p> <p>Properties Nillable</p> <p>Description The application name in English. For example, Salesforce Internal Application or Microsoft SOAP Toolkit.</p>
AuthServiceId	<p>Type reference</p> <p>Properties Nillable</p> <p>Description Refers to the AuthenticationServiceId field on the LoginHistory object. For example, you can use this field to identify the SAML or authentication provider configuration with which the user logged in.</p>
Browser	<p>Type string</p> <p>Properties Nillable</p> <p>Description The browser name and version if known. If no browser or version number is available, "Unknown" is returned. Product names are in English.</p>
CipherSuite	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The TLS cipher suite used for the login. Values are OpenSSL-style cipher suite names, with hyphen delimiters, for example, AES128-GCM-SHA256. Available in API version 37.0 and later. Valid values are:</p> <ul style="list-style-type: none"> • AES128-GCM-SHA256 • AES128-SHA • AES128-SHA256 • AES256-GCM-SHA384 • AES256-SHA • AES256-SHA256 • DES-CBC3-SHA • DHE-RSA-AES128-GCM-SHA256

Field	Details
	<ul style="list-style-type: none"> • DHE-RSA-AES128-SHA • DHE-RSA-AES256-GCM-SHA384 • DHE-RSA-AES256-SHA • DHE-RSA-DES-CBC3-SHA • ECDH-ECDSA-AES128-GCM-SHA256 • ECDH-ECDSA-AES128-SHA256 • ECDH-ECDSA-AES256-GCM-SHA384 • ECDH-ECDSA-AES256-SHA384 • ECDH-RSA-AES128-GCM-SHA256 • ECDH-RSA-AES128-SHA256 • ECDH-RSA-AES256-GCM-SHA384 • ECDH-RSA-AES256-SHA384 • ECDHE-ECDSA-AES128-GCM-SHA256 • ECDHE-ECDSA-AES128-SHA256 • ECDHE-ECDSA-AES256-GCM-SHA384 • ECDHE-ECDSA-AES256-SHA384 • ECDHE-RSA-AES128-CBC-SHA • ECDHE-RSA-AES128-GCM-SHA256 • ECDHE-RSA-AES128-SHA256 • ECDHE-RSA-AES256-CBC-SHA • ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-RSA-AES256-SHA384 • ECDHE-RSA-DES-CBC3-SHA • Unknown
ClientVersion	<p>Type string</p> <p>Properties Nillable</p> <p>Description English-only. The version number of the login client. If no version number is available, "Unknown" is returned.</p>
EventDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The login time of the specified event. For example, 2013-01-01T03:01:01Z. Seconds are the most granular setting.</p>

Field	Details
LoginGeoId	Type reference
	Properties Nillable
	Description The Salesforce ID of the LoginGeo object associated with the login user's IP address.
LoginHistoryId	Type reference
	Properties Nillable
	Description Tracks a user session so you can correlate user activity with a particular login instance.
LoginType	Type string
	Properties Nillable, Restricted picklist
	Description The event's type of login. For example, "Application."
LoginUrl	Type string
	Properties Nillable
	Description The URL of the login host. For example, <i>yourInstance</i>.salesforce.com .
NetworkId	Type reference
	Properties Nillable
	Description The ID of the community that the user is logging in to. Available in API version 37.0 and later, if Salesforce Communities is enabled for your org.
Platform	Type string
	Properties Nillable

Field	Details
	<p>Description The platform name and version that are used during the login event. If no platform name is available, "Unknown" is returned. Platform names are in English.</p>
SourceIp	<p>Type string</p> <p>Properties Nillable</p> <p>Description The source IP address of the client logging in. For example, 126.7.4.2.</p>
Status	<p>Type string</p> <p>Properties Nillable</p> <p>Description The login status is in English. "Success" is returned when successful. Otherwise, an error message is returned.</p>
TlsProtocol	<p>Type picklist</p> <p>Properties Nillable, Restricted picklist</p> <p>Description The TLS protocol version used for the login. Available in API version 37.0 and later. Valid values are:</p> <ul style="list-style-type: none">• TLS 1.0• TLS 1.1• TLS 1.2• Unknown
UserId	<p>Type ID</p> <p>Properties Nillable</p> <p>Description The user's unique ID. For example, 00500000000123.</p>
Username	<p>Type string</p> <p>Properties Nillable</p>

Field	Details
Description	
The username in the format of <code>user@company.com</code> .	

Working with AdditionalInfo

`AdditionalInfo` enables you to extend the login event with custom data that can be queried later. For example, you can capture a correlation ID when a user logs in from an external system that shares that unique ID. This process enables tracking logins across systems. To store data with `LoginEvent`, begin all `AdditionalInfo` field names with `x-sfdc-addinfo-{fieldname}`. For example, a valid field assignment is `x-sfdc-addinfo-correlation_id = ABC123` where `x-sfdc-addinfo-correlation_id` is the field name and `ABC123` is the field value.

When defining field names, note the following:

- `x-sfdc-addinfo-` is case-insensitive. `x-sfdc-addinfo-{field name}` is the same as `X-SFDC-ADDINFO-{FIELD NAME}`.
- Fields can contain only alphanumeric and “_” (underscore) characters.
- Field names must be between 2 and 29 characters in length, excluding `x-sfdc-addinfo-`.
- Field names that don’t start with `x-sfdc-addinfo-` are ignored.
- Names that contain invalid characters after `x-sfdc-addinfo-` are ignored.
- Only the first 30 valid field names are stored in `AdditionalInfo`. Field names are not necessarily stored in the same order in which they were passed to authentication.

When determining field values, keep the following in mind:

- You can’t use existing API field names as `AdditionalInfo` names in the HTTP header. If the `AdditionalInfo` name conflicts with an object’s API name, the field value isn’t stored. For example, the HTTP header `X-SFDC-ADDINFO-UserId='abc123'` doesn’t get stored in `AdditionalInfo`.
- Additional field values can contain only alphanumeric, “_” and “-” characters.
- Field values must be 255 characters in length or fewer. If a field value exceeds 255 characters, only the first 255 characters are stored and the rest are truncated.
- Field values that contain invalid characters are saved with a field header of Empty String ("").
- Only the first 30 valid field names are stored in the `AdditionalInfo` field. They are not guaranteed to be stored in the same order that they were passed into the authentication.
- When `AggregationFieldName` is `SourceIp`, you can’t filter on `AggregationFieldValue` if its value is `Salesforce.com IP`.

How to Pass Additional Information by Using HTTP with cURL

Here’s an example of passing additional information via the command line.

```
curl https://yourInstance.salesforce.com/services/oauth2/token -d "grant_type=password"
-d
"client_id=3MVG9PhR6g6B7ps4RF_kNPoWSxVQstrazijsE8njPtkpUzVPPffzy8
jIoRE6q9rPznNt1sqbP9ob8kUfMjXXX" -d "client_secret=4180313776440635XXX" -d
"username=user@company.com" -d "password=123456" -H "X-PrettyPrint:1" -H
```

```
"x-sfdc-addinfo-correlationid:  
d18c5a3f-4fba-47bd-bbf8-6bb9a1786624"
```

How to Pass Additional Information in Java

Here's an example of passing additional information in Java.

```
//adding additional info headers ..
Map<String, String> httpHeaders = new HashMap<String, String>();
httpHeaders.put("x-sfdc-addinfo-fieldname1" /* additional info field*/ ,
" d18c5a3f-4fba-47bd-bbf8-6bb9a1786624" /* value*/);
httpHeaders.put("x-sfdc-addinfo-fieldname2" /* additional info field*/ ,
" d18c5a3f-4fba-47bd-bbf8-6bb9a1786624" /* value*/);
ConnectorConfig config = new ConnectorConfig();
config.setUsername(userId);
config.setPassword(passwd);
config.setAuthEndpoint(authEndPoint);
config.setProxy(proxyHost, proxyPort);
//setting additional info headers
for (Map.Entry<String, String> entry : httpHeaders.entrySet()) {
config.setRequestHeader(entry.getKey(), entry.getValue());
}
// Set the username and password if your proxy must be authenticated
9
LoginEvent
config.setProxyUsername(proxyUsername);
config.setProxyPassword(proxyPassword);
try {
EnterpriseConnection connection = new EnterpriseConnection(config);
// etc.
} catch (ConnectionException ce) {
ce.printStackTrace();
}
```

SOQL Usage

Currently, the only supported SOQL function on LoginEvent is WHERE, and you can only use comparison operators (=, <, >, <=, and >=) on the final expression in a WHERE clause. The != operator isn't supported.



Note: Date functions such as `convertTimezone()` aren't supported—for example, `SELECT CALENDAR_YEAR(EventDate), Count(Id) FROM LoginEvent GROUP BY CALENDAR_YEAR(EventDate)` returns an error. You can use date literals in your queries and some date and date/time functions like TODAY, YESTERDAY, and LAST_n_DAYS:1. However, these functions use comparison operators behind the scenes. This means you can only use them in the final expression of a WHERE clause.

LoginEvent allows filtering over two ordered fields: `EventDate` and `Id`. There's a catch here; your query won't work unless you use the correct order and combination of these fields. The following list provides some examples of valid and invalid queries:

- **Unfiltered**

- **Valid**—contains no WHERE clause, so no special rules apply.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent
```

- **Filtered on EventDate**

- **Valid**—you can filter solely on EventDate, but single filters on other fields fail. You can also use a comparison operator in this query type.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent  
WHERE EventDate<=2014-11-27T14:54:16.000Z
```

- **Valid**—you can filter on EventDate using date literals.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent  
WHERE EventDate<=TODAY
```

- **Filtered on EventDate and Id**

- **Valid**—successful queries on LoginEvent filter over both fields.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent  
WHERE EventDate=2014-11-27T14:54:16.000Z and Id='1HBD00000001N6EOAU'
```

- **Valid**—successful queries on LoginEvent with EventDate and standard date literals.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent  
WHERE EventDate=TODAY and Id='1HB0D0000000kJDWAY'
```

- **Invalid**—filtering only on EventDate with <= or >= operator and Id field isn't supported.

```
SELECT Application, Browser, EventDate, Id, LoginUrl, UserId  
FROM LoginEvent  
WHERE EventDate<=2014-11-27T14:54:16.000Z and Id='1HBD00000001N6EOAU'
```

LoginGeo

Represents the geographic location of the user's IP address for a login event. Due to the nature of geolocation technology, the accuracy of geolocation fields (for example, country, city, postal code) may vary. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Only users with Manage Users permissions can access this object.

Fields

Field	Details
City	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The city where the user's IP address is physically located. This value is not localized.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The country where the user's IP address is physically located. This value is not localized.</p>
CountryIso	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ISO 3166 code for the country where the user's IP address is physically located. For more information, see Country Codes - ISO 3166</p>
Latitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The latitude where the user's IP address is physically located.</p>
LoginTime	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description Time of the login attempt, in GMT time zone.</p>
Longitude	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description The longitude where the user's IP address is physically located.</p>
PostalCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The postal code where the user's IP address is physically located. This value is not localized.</p>
Subdivision	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the subdivision where the user's IP address is physically located. In the U.S., this value is usually the state name (for example, Pennsylvania). This value is not localized.</p>

Usage

The API allows you to do many powerful queries. A few examples are:

Sample Query	Query String
Query showing the country for a login event, where Id=LoginGeoId from AuthSession	SELECT Country FROM LoginGeo WHERE Id = 'OLE#####'
Query showing the city and postal code for a login event, where Id=LoginGeoId from LoginHistory	SELECT City, PostalCode FROM LoginGeo WHERE Id = 'OSO#####'

LoginHistory

Represents the login history for all successful and failed login attempts for organizations and enabled portals. This object is available in API version 21.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

With one exception, only users with Manage Users permissions can access this object. The exception is that, in API version 37.0 and later, all users can retrieve their own login history records.

Fields

Field	Details
ApiType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Indicates the API type, for example Soap Enterprise. Label is API Type.</p>
APIVersion	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Displays the API version used by the client. Label is API Version.</p>
Application	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The application used to access the organization. Label is Application.</p>
AuthenticationServiceId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 18-character ID for an authentication service for a login event. For example, you can use this field to identify the SAML or authentication provider configuration with which the user logged in. This field is available in API version 34.0 and later. Label is Authentication Service Id.</p>
Browser	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
	Description
	The current browser version. Label is Browser .
ClientVersion	Type string
	Properties
	Group, Nillable, Sort
	Description
	Version of the API client. Label is Client Version .
CipherSuite	Type picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	The TLS cipher suite used for the login. Values are OpenSSL-style cipher suite names, with hyphen delimiters. For more information, see OpenSSL Cryptography and SSL/TLS Toolkit . This field is available in API version 37.0 and later.
CountryIso	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The ISO 3166 code for the country where the user's IP address is physically located. For more information, see Country Codes - ISO 3166 . This field is available in API version 37.0 and later.
LoginGeoId	Type reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	The 18-character ID for the record of the geographic location of the user for a successful or unsuccessful login event. Due to the nature of geolocation technology, the accuracy of geolocation fields (for example, country, city, postal code) can vary. This field is available in API version 34.0 and later.
LoginTime	Type dateTime
	Properties
	Filter, Nillable, Sort

Field	Details
	<p>Description Time zone is based on GMT. Label is Login Time.</p>
LoginType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of login, for example, Application, OAuth, or SAML. Label is Login Type.</p>
LoginUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL from which the login request is coming. Label is Login URL.</p>
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the community that the user is logging in to. This field is available in API version 31.0 and later, if Salesforce Communities is enabled for your organization.</p>
Platform	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Operating system on the login machine. Label is Platform.</p>
SourceIp	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description IP address of the machine from which the login request is coming. The address can be an IPv4 or IPv6 address in API version 23.0 or later. In API version 22.0 or earlier, the address is an IPv4 address, and IPv6 addresses are null. Label is Source IP.</p>

Field	Details
Status	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Displays the status of the attempted login. Status is either success or a reason for failure. Label is Status.</p>
TlsProtocol	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The TLS protocol used for the login. Possible values are:</p> <ul style="list-style-type: none"> • TLS 1.0 • TLS 1.1 • TLS 1.2 • Unknown <p>This field is available in API version 37.0 and later.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user logging in. Label is User ID.</p>

Usage

Not all fields are filterable. You can only filter on the following fields:

- AuthenticationServiceId
- CipherSuite
- CountryIso
- Id
- LoginTime
- LoginType
- LoginUrl
- NetworkId
- TlsProtocol

- `UserId`

The API allows you to do many powerful queries. A few examples are:

Sample Query	Query String
Simple query showing UserId & LoginTime for each user	<code>SELECT UserId, LoginTime from LoginHistory;</code>
Query showing logins only after a specified date and time	<code>SELECT UserId, LoginTime from LoginHistory WHERE LoginTime > 2010-09-20T22:16:30.000Z;</code>
Query showing logins for a specific time interval	<code>SELECT UserId, LoginTime from LoginHistory WHERE LoginTime > 2010-09-20T22:16:30.000Z AND LoginTime < 2010-09-21T22:16:30.000Z;</code>
Query showing the authentication service for a SAML login event, where Id=AuthenticationServiceId from LoginHistory	<code>SELECT name, issuer, samlVersion FROM SamlSsoConfig WHERE Id = 'OLE#####'</code>
Query showing the authentication service for an authentication provider login event, where Id=AuthenticationServiceId from LoginHistory	<code>SELECT Type, DeveloperName FROM AuthProvider WHERE Id = 'OSO#####'</code>

LoginIp

Represents a validated IP address. This object is available in version 28.0 and later.

Supported Calls

`describeSObjects(), delete(), query(), retrieve()`

Fields

Field	Details
<code>ChallengeMethod</code>	<p>Type picklist</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The challenge method used to confirm the user's identity. Possible values include the following.</p> <ul style="list-style-type: none"> • Email • SMS • TOTP_CHOICE: The user chooses two-factor authentication. • TOTP_ONLY: The user is required to use two-factor authentication.

Field	Details
ChallengeSentDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time that the user was authenticated.</p>
IsAuthenticated	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has already been authenticated.</p>
SourceIp	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The IP address the user logged in from.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the user associated with this item.</p>

Usage

At every login, the IP address of the login request is checked against the validated IP addresses using LoginIp. A match means the login IP address is a known IP address. If there's no match, the address is unknown, and the user is asked to confirm their identity.

LookedUpFromActivity

This read-only object is displayed as a related list on an activity record (an event or a task); the list contains records that have custom lookup relationships from the activity to another object. This object is not queryable.

Supported Calls

`describeSObjects()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the related account, which is determined as follows:</p> <ul style="list-style-type: none"> • The account associated with the <code>WhatId</code>, if it exists; or • The account associated with the <code>WhoId</code>, if it exists; otherwise • <code>null</code> <p>For information on IDs, see Field Types</p>
ActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates one of the following:</p> <ul style="list-style-type: none"> • The due date of a task • The date of an event if <code>IsAllDayEvent</code> is set to <code>true</code> <p>This field has a time stamp that is always set to midnight in the Universal Time Coordinated (UTC) time zone. The time stamp doesn't represent the time of the activity; don't attempt to alter it to accommodate time zone differences. Label is Date.</p>
ActivitySubtype	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Provides standard subtypes to facilitate creating and searching for specific activity subtypes. This field isn't updateable.</p> <p>ActivitySubtype values:</p> <ul style="list-style-type: none"> • Task • Email • Call • Event

Field Name	Details
ActivityType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents one of the following values: Call, Meeting, or Other. Label is Type.</p>
CallDisposition	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the result of a given call; for example, "we'll call back," or "call unsuccessful." Limit is 255 characters.</p>
CallDurationInSeconds	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Duration of the call in seconds.</p>
CallObject	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of a call center. Limit is 255 characters.</p>
CallType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of call being answered: Inbound, Internal, or Outbound.</p>
Description	<p>Type textarea</p> <p>Properties Nillable</p>

Field Name	Details
	Description
	<p>Contains a description of the event or task. Limit is 32 KB.</p>
DurationInMinutes	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>Indicates the duration of the event or task.</p>
EndDateTime	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	<p>Indicates the end date and time of the event or task. Available in versions 27.0 and later. This field is optional, depending on the following:</p>
	<ul style="list-style-type: none"> • If <code>IsAllDayEvent</code> is true, you can supply a value for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are <code>null</code>, the duration defaults to one day. • If <code>IsAllDayEvent</code> is false, a value must be supplied for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time.
IsAllDayEvent	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	<p>If the value of this field is set to <code>true</code>, then the activity is an event spanning a full day, and the <code>ActivityDate</code> defines the date of the event. If the value of this field is set to <code>false</code>, then the activity may be an event spanning less than a full day, or it may be a task. Label is <code>All-Day Event</code>.</p>
IsClosed	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort

Field Name	Details
	<p>Description Indicates whether a task is closed; value is always <code>false</code>. This field is set indirectly by setting <code>Status</code> on the task—each picklist value has a corresponding <code>IsClosed</code> value. Label is <code>Closed</code>.</p>
<code>IsHighPriority</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates a high-priority task. This field is derived from the <code>Priority</code> field.</p>
<code>IsReminderSet</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a reminder is set for an activity (<code>true</code>) or not (<code>false</code>).</p>
<code>IsTask</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If the value of this field is set to <code>true</code>, then the activity is a task; if the value is set to <code>false</code>, then the activity is an event. Label is <code>Task</code>.</p>
<code>IsVisibleInSelfService</code>	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If the value of this field is set to <code>true</code>, then the activity can be viewed in the self-service portal. Label is <code>Visible in Self-Service</code>.</p>
<code>Location</code>	<p>Type <code>string</code></p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the activity is an event, then this field represents the location of the event. If the activity is a task, then the value is <code>null</code>.</p>

Field Name	Details
OwnerId	Type reference Properties Filter, Group, Nillable, Sort
	Description Indicates the ID of the user who owns the activity.
Priority	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Sort Description Indicates the priority of a task, such as high, normal, or low.
ReminderDateTime	Type dateTime Properties Filter, Nillable, Sort Description Represents the time at which a reminder is scheduled to fire if <code>IsReminderSet</code> is set to <code>true</code> . If <code>IsReminderSet</code> is set to <code>false</code> , then either the user has deselected the reminder checkbox in the user interface or the reminder has already fired at the time indicated by the value.
StartTime	Type dateTime Properties Filter, Nillable, Sort Description Indicates the start date and time of the event. Available in versions 13.0 and later. The <code>StartTime</code> field contains the event start date. However, if the event's <code>IsAllDayEvent</code> flag is set to <code>true</code> (indicating an all-day event), then the time stamp in the <code>StartTime</code> field is always set to midnight in the Coordinated Universal Time (UTC) time zone. Don't attempt to alter the time stamp to account for any time zone differences. If the event's <code>IsAllDayEvent</code> flag is set to <code>false</code> , then you must translate the time portion of the time stamp in the <code>StartTime</code> field to or from a local time zone for the user or the application, as appropriate, and the translation must be in the Coordinated Universal Time (UTC) time zone. If this field has a value, then <code>ActivityDate</code> and <code>ActivityDateTime</code> either must be <code>null</code> or must match the value of this field.

Field Name	Details
Status	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Sort</p> <p>Description Indicates the current status of a task, such as in progress or complete. Each predefined status field sets a value for <code>IsClosed</code>.</p>
Subject	<p>Type combobox</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Contains the subject of the task or event.</p>
WhatId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The <code>WhatId</code> represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. <code>WhatIds</code> are polymorphic. Polymorphic means a <code>WhatId</code> is equivalent to the ID of a related object. The label is <code>Related To ID</code>.</p>
WhoId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The <code>WhoId</code> represents a human such as a lead or a contact. <code>Whoids</code> are polymorphic. Polymorphic means a <code>WhoId</code> is equivalent to a contact's ID or a lead's ID. The label is <code>Name ID</code>.</p>

Usage

Query activities related to an object

1. Optionally, issue a describe call against the object whose activities you wish to query, to get a suggestion of the correct SOQL to use.

2. Issue a SOQL relationship query with a main clause that references the object, and an inner clause that references the activity custom lookup relationship; for example:

```
SELECT id, name,
    (SELECT id, subject from sponsoredact __r)
FROM Contact
```

In this example `sponsoredact __r` is a user defined relationship list.

The user interface enforces sharing rules, filtering out related-list items that a user doesn't have permission to see.

The following restrictions on users who don't have "View All Data" permission help prevent performance issues:

- In the main clause of the relationship query, you can reference only one record. For example, you can't filter on all records where the account name starts with 'A'; instead, you must reference a single account record.
- In the inner clause of the query, you can't use `WHERE`.
- In the inner clause of the query, you must specify a limit of 500 or fewer on the number of rows that are returned in the list.
- You must sort on `ActivityDate` in descending order and `LastModifiedDate` in descending order; you can display nulls last. For example: `ORDER BY ActivityDate DESC NULLS LAST, LastModifiedDate DESC`.

MacroInstruction

Represents an instruction in a macro. An instruction can specify the object that the macro interacts with, the context or publisher that the macro works within, the operation or action that the macro performs, and the target of the macro's actions. It is a useful way to programmatically define instructions, instead of using the macro widget in the console.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
<code>MacroId</code>	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the macro that contains this instruction.</p>
<code>Name</code>	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description Name of the instruction.</p>

Field Name	Details
Operation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The action that the macro instruction performs.</p>
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Order of this instruction in the macro.</p>
Target	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Object in the case feed that is the target of the operation. For example, the target might be the active case tab or a quick action.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Value of a field. If the operation is Select, then the value is null, because the operation selects the object on which the macro performs an action. An instruction can contain both a <code>Value</code> field and a <code>ValueRecord</code> field, but only one of these fields can have a value. The other field value must be null.</p>
ValueRecord	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the value or record. The <code>ValueRecord</code> can be either a value or a record, but not both. An instruction can contain both a <code>Value</code> field and a <code>ValueRecord</code> field, but only one of these fields can have a value. The other field value must be null.</p>

MailmergeTemplate

Represents a mail merge template (a Microsoft Word document) used for performing mail merges for your organization.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`,
`undelete()`, `update()`, `upsert()`

Special Access Rules

- All users can view this object, but you need the “Customize Application” permission to modify it.
- Customer Portal users can’t access this object.

Fields

Field	Details
Body	Type base64 Properties Create Description Required. Microsoft Word document to use as a mail merge template. Due to limitations with Microsoft Word mail merge templates, your client application can specify the Body field when creating these records, but not when updating them. Limit: 5 MB.
BodyLength	Type int Properties Filter, Group, Nillable, Sort Description Length of the Microsoft Word document.
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Required. Text description of this mail merge template. Limit: 255 characters.
Filename	Type string

Field	Details
	<p>Properties Create, Filter, Group, Sort</p> <p>Description Required. File name of the Microsoft Word document that was uploaded as a mail merge template. Limit: 255 characters in length.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LastUsedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when this MailmergeTemplate was last used.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of this mail merge template.</p>
SecurityOptionsAttachmentHasFlash	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Required. True if Flash Injection was detected in the attachment.</p>
SecurityOptionsAttachmentHasXSSThreat	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Required. True if a cross site scripting threat was detected in the attachment.</p>

Field	Details
SecurityOptionsAttachmentScannedforFlash	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Required. True if the attachment has been scanned for Flash Injection.</p>
SecurityOptionsAttachmentScannedForXSS	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Required. True if the attachment has been scanned for a cross site scripting threat.</p>

Usage

Use this object to manage mail merge templates for your organization.

SEE ALSO:

[Object Basics](#)

MatchingRule

Represents a matching rule that is used to identify duplicate records. This object is available in API version 33.0 and later.

A matching rule compares field values to determine whether a record is similar enough to existing records to be considered a duplicate. For example, a matching rule can specify that if the `Email` and `Phone` values of two records match exactly, the records might be duplicates. Your organization uses matching rules with duplicate rules to define what happens when duplicates are identified.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
BooleanFilter	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Specifies filter logic conditions. For more information on filter logic, see "Getting the Most Out of Filter Logic" in the Salesforce Help.</p>
Description	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The description of the matching rule.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The developer name for the matching rule.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language selected for your organization.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The name of the matching rule.</p>
MatchEngine	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The match engine used by the matching rule.</p>

Field Name	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix for matching rules for your organization.</p>
RuleStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Required. The activation status of the matching rule. Values are:</p> <ul style="list-style-type: none"> • <i>Inactive</i> • <i>Deactivating</i> • <i>DeactivationFailed</i> • <i>Active</i> • <i>Activating</i> • <i>ActivationFailed</i> <p> Important: The only valid values you can declare when deploying a package are <i>Active</i> and <i>Inactive</i>.</p>
ObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The object for the matching rule.</p>

Usage

Use the Salesforce API to retrieve and view details about MatchingRule and MatchingRuleItem. Use the Salesforce Metadata API to create, update, or delete these objects.

SEE ALSO:

[MatchingRuleItem](#)
[DuplicateRule](#)
[DuplicateResult](#)

MatchingRuleItem

Represents criteria used by a matching rule to identify duplicate records. This object is available in API version 33.0 and later.

A matching rule item determines which field the matching rule uses to identify a duplicate record. It also determines the method used to compare value that two records have for the field. For example, a matching rule item might specify that the `Email` field values of two records must match exactly in order for the records to be considered duplicates.

When a matching rule has multiple matching rule items, it means that multiple fields must match in order for the records to be identified as duplicates.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
BlankValueBehavior	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Specifies how blank fields affect whether the fields being compared are considered matches. Valid values are:</p> <ul style="list-style-type: none"> • <code>MatchBlanks</code> • <code>NullNotAllowed</code> (default)
Field	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates which field to compare when determining if a record is similar enough to an existing record to be considered a match.</p>
MatchingMethod	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Defines how the fields are compared. Choose between the exact matching method and various fuzzy matching methods. Valid values are:</p> <ul style="list-style-type: none"> • <code>Exact</code>

Field Name	Details
	<ul style="list-style-type: none">• <i>FirstName</i>• <i>LastName</i>• <i>CompanyName</i>• <i>Phone</i>• <i>City</i>• <i>Street</i>• <i>Zip</i>• <i>Title</i>
For details on each matching method, see “Matching Methods Used with Matching Rules” in the Salesforce Help.	
MatchingRuleId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID for the matching rule.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The order of the matching rule items for a matching rule.</p>

Usage

Use the Salesforce SOAP API to retrieve and view details about MatchingRule and MatchingRuleItem. Use the Salesforce Metadata API to create, update, or delete these objects.

SEE ALSO:

[MatchingRule](#)
[DuplicateRule](#)
[DuplicateResult](#)

MetadataPackage

Represents a managed or unmanaged package that has been developed in the org you’re logged in to.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The name of the package.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description For managed packages, this field is the namespace prefix assigned to the package. For unmanaged packages, this field is blank.</p>

Usage

Here are examples of the types of API queries you can perform.

Query	String
Show all managed and unmanaged packages in the org	<code>SELECT Name, NamespacePrefix FROM MetadataPackage</code>
Show only managed packages in the org	<code>SELECT Name, NamespacePrefix FROM MetadataPackage WHERE NamespacePrefix <> ''</code>

MetadataPackageVersion

Represents a package version (managed or unmanaged) that has been uploaded from the org you're logged in to.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
BuildNumber	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The build number of the version. For example, if you upload two beta versions, they have build numbers 1 and 2. Then, when you upload a non-beta version, the build number is 3. When you upload a new version, the build number resets to 1.</p>
MajorVersion	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first number in a package version number. A version number either has an <code>x.y</code> format or an <code>x.y.z</code> format. The <code>x</code> represents the major version, <code>y</code> the minor version, and <code>z</code> the patch version.</p>
MetadataPackageId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 18-character package ID starting with 033.</p>
MinorVersion	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The second number in a package version number. A version number either has an <code>x.y</code> format or an <code>x.y.z</code> format. The <code>x</code> represents the major version, <code>y</code> the minor version, and <code>z</code> the patch version.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p>

Field Name	Details
	Description
PatchVersion	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The third number in a package version number, if present. A version number either has an <code>x.y</code> format or an <code>x.y.z</code> format. The <code>x</code> represents the major version, <code>y</code> the minor version, and <code>z</code> the patch version.</p>
ReleaseState	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description If the package version is a beta version, the value is <code>Beta</code>. Otherwise, the value is <code>Released</code>.</p>

Usage

Here are examples of the types of API queries you can perform.

Query	String
Get all package versions for the package that has a <code>MetadataPackageID</code> of <code>033D00000001xQ1IAI</code>	<code>SELECT Id, Name, ReleaseState, MajorVersion, MinorVersion, PatchVersion FROM MetadataPackageVersion WHERE MetadataPackageId = '033D00000001xQ1IAI'</code>
Get the package version for the package with a specific <code>MetadataPackageID</code> and a major version greater than 1	<code>SELECT Id FROM MetadataPackageVersion WHERE MetadataPackageId = '033D00000001xQ1IAI' AND MajorVersion > 1</code>
Get released package versions for the package with a specific <code>MetadataPackageID</code>	<code>SELECT Id FROM MetadataPackageVersion WHERE MetadataPackageId = '033D00000001xQ1IAI' AND ReleaseState = 'Released'</code>

Java Code Sample

Suppose you want to push version 3.4.6 of your package to all orgs. Let's write some code to identify the orgs eligible for the upgrade. This example demonstrates how to generate the list of subscriber orgs eligible to be upgraded to version 3.4.6 of a package.

This code sample uses the Force.com Web Services Connector (WSC).

```
// Finds all Active subscriber orgs that have the package installed
String PACKAGE_SUBSCRIBER_ORG_KEY_QUERY = "Select OrgKey from PackageSubscribers where
OrgStatus = 'Active' and InstalledStatus = 'I'";

// Finds all MetadataPackageVersions lower than the version given, including the list
// of subscribers for each version
String METADATA_PACKAGE_VERSION_QUERY = "Select Id, Name, ReleaseState, (%s) from"
+ " MetadataPackageVersion where MetadataPackageId = '%s' AND ReleaseState = 'Released'"

+ " AND (MajorVersion < 3 OR (MajorVersion = 3 and MinorVersion < 4))"
+ " OR (MajorVersion = 3 and MinorVersion = 4 and PatchVersion < 6))";

// conn is an EnterpriseConnection instance initialized with a ConnectionConfig object
// representing a connection to the developer org of the package
QueryResult results = conn.query(String.format(METADATA_PACKAGE_VERSION_QUERY,
PACKAGE_SUBSCRIBER_ORG_KEY_QUERY));

// This list will hold all of the PackageSubscriber objects that are eligible for upgrade
// to the given version
List<PackageSubscriber> subscribers = new ArrayList<>();
for (SObject mpvso : results.getRecords()) {

    // Cast the sObject to a MetadataPackageVersion
    MetadataPackageVersion mpv = (MetadataPackageVersion) mpvso;

    // Add subscribers to our list
    if (mpv.getPackageSubscribers() != null) {
        for (SObject pss : mpv.getPackageSubscribers().getRecords()) {
            subscribers.add((PackageSubscriber) pss);
        }
    }
}
```

Next Step

Create a push request using PackagePushRequest.

Metric

The Metric object represents the components of a goal metric such as its name, metric type, and current value.

Supported Calls

create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()

Fields

Field Name	Details
CompletionDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The completion date of the metric.</p>
CurrentValue	<p>Type double</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description The current value of the metric.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the metric. The maximum length is 65,535 characters.</p>
DueDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The due date of the metric.</p>
GoalId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the goal the metric is related to.</p>
InitialValue	<p>Type double</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The initial value of the metric.</p>
IsCompletionMetric	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read only. If <code>true</code>, the metric measures whether or not the metric is finished. If <code>false</code>, the metric measures how much is finished compared to a targeted value.</p>
LastComment	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A comment that provides more context about the metric, such as its status or progress. The maximum length is 255 characters.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp that indicates when a user last viewed a record that is related to this metric.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp that indicates when a user last viewed this metric. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description The name of the metric.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns the metric.</p>
Progress	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The overall progress of the metric.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the related record type.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The start date of the metric.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The status of the metric. Possible values include:</p> <ul style="list-style-type: none">• Not Started• On Track• Behind• Critical

Field Name	Details
	<ul style="list-style-type: none"> Completed Postponed Canceled Not Completed
TargetValue	<p>Type double</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description The target value of the metric.</p>
Weight	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The weight of the metric. The sum of the weights should equal 100%.</p>

MetricDataLink

The link between the metric and the data source, such as a report.

Supported Calls

```
create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(),
undelete(), update(), upsert()
```

Fields

Field Name	Details
DatasourceFieldName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The field name of the data source, such as a report summary field.</p>

Field Name	Details
DataSourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the data source.</p>
LastSynchronizationTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The last time the data was synchronized.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The name given to the data link record.</p>
TargetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the metric that the data is linked to.</p>

MetricDataLinkHistory

This read-only object contains historical information about changes that have been made to the MetricDataLink object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
MetricDataLinkId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the MetricDataLink.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

MetricFeed

Represents a single feed item in the feed displayed on a Metric record.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The numbers of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is ContentPost. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Available in API version 36.0 and earlier only. Group, Nillable, Sort</p> <p>Description Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Available in API version 36.0 and earlier only. Group, Nillable, Sort</p> <p>Description Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then <code>InsertedById</code> is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p>

Field Name	Details
	<ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the metric record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>

Field Name	Details
	<p>Properties Group, Nillable, Sort</p>
	<p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type string</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p>
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API.</p>
	<ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file.

Field Name	Details
	<ul style="list-style-type: none"> • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated.

MetricHistory

This read-only object contains historical information about changes that have been made to the Metric object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
MetricId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the Metric.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

MetricsDataFile

Represents a data file containing usage metrics on all installations of a managed package in a Salesforce instance. This object is available in API version 30.0 and later.

Supported Calls

`query()`, `delete()`

Fields

Field Name	Details
MetricsDataFile	<p>Type base64</p> <p>Properties Filter, Query, Sort</p> <p>Description A text file containing the usage data encoded in Base 64.</p>

Field Name	Details
MetricsDataFileType	Type string
Properties	Filter, Query, Sort
Description	The format of the data file. Currently, the only allowed value is <code>text/csv</code> .
MetricsDataFileLength	Type int
Properties	Filter, Query, Sort
Description	The size of the data file in bytes.
MetricsRunDate	Type dateTime
Properties	Filter, Query, Sort
Description	The date when the usage metrics collection job was run.
MetricsEndDate	Type dateTime
Properties	Filter, Query, Sort
Description	The end time and date for the data collection.
MetricsStartDate	Type dateTime
Properties	Filter, Query, Sort
Description	The start time and date for the data collection.
MetricsType	Type picklist
Properties	Filter, Query, Sort

Field Name	Details
	Description
	The type of data being collected. The possible values are <code>CustomObject</code> and <code>Visualforce</code> .
NamespacePrefix	Type
	string
	Properties
	Filter, Query, Sort
	Description
	The namespace prefix of the package for which data is being collected.
SendingInstance	Type
	string
	Properties
	Filter, Query, Sort
	Description
	The server instance from which this data was collected, for example, "na8."

Usage

Use this object to access customer usage metrics for a managed package. Each record contains one day's data, on either custom objects or Visualforce pages, for all organizations in a Salesforce instance that have the package installed. The following data is collected each day.

- **Custom objects** — the number of records stored in each custom object.
- **Visualforce pages** — the number of times each Visualforce page was accessed, the number of unique users who accessed it, and the average loading time (in milliseconds).

The custom objects data is a snapshot that reflects the state of the organization at the time the database was sampled, while the Visualforce data covers usage over a 24-hour period.

This feature is intended for API access only. The owner of the package must write a secondary process to retrieve the metrics data from the reporting organization, and export it to another system for analysis.

The usage metrics data for all production organizations in a given instance is merged and written into a text file, in a specified format, once a day. If an instance doesn't have any organizations with the package installed or any organizations that accessed Visualforce pages in the package, a blank record is created for that day, with `MetricsDataFileLength` set to zero.

In a record for custom objects, each row of the text file contains usage data in the following order.

- Organization ID
- Organization name
- Organization edition
- Organization status
- Name of the custom object
- Number of records of the custom object on the specified day

The custom object count is a snapshot captured once each day. Here's a section of a sample data file for custom objects. It shows there were 3500 and 1500 records in the Alpha and Beta custom objects, respectively, in the specified customer organization on the specified day.

```
"00Dxx0000001gbk","org1","Enterprise Edition","TRIAL","Alpha", "3500"
"00Dxx0000001gbk","org1","Enterprise Edition","TRIAL","Beta", "1500"
```

In a record for Visualforce pages, each row of the text file contains usage data in the following order.

- Organization ID
- Organization name
- Organization edition
- Organization status
- Package version number
- Name of the Visualforce page
- Number of times the page was accessed
- Number of unique users who accessed the page
- Average loading time of the page, in milliseconds

The Visualforce counts for each organization measure the number of times the page was viewed in the duration between the start and end times. Here's a section of a sample data file for Visualforce pages.

```
"00Dxx0000001gbk","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f1","1","1","66.0"
"00Dxx0000001gbk","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f2","1","1","128.0"
"00Dxx0000001gbk","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f3","1","1","107.0"
"00Dxx0000001gbf","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f1","5","1","73.6"
"00Dxx0000001gbf","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f2","1","1","72.0"
"00Dxx0000001gbf","org1","Enterprise Edition","TRIAL","1.0","/apex/gm12_f3","7","1","50.8"
```

MetricShare

Represents a sharing entry on a Metric object.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description The user's or group's level of access to the metric. The possible values are:</p> <ul style="list-style-type: none">• Read Only• Read/Write• Owner <p>This field must be set to an access level that is higher than the organization's default access level for metrics.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the Metric object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none">• Owner: User is the owner of the Metric or is in a user role above the Metric owner in the role hierarchy.• Manual: User or group has access because a user with "All" access manually shared the Metric with the user or group.• Rule: User or group has access via a Metric sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the Metric. This field can't be updated.</p>

MilestoneType

Represents a milestone (required step in a customer support process). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description A description of the milestone.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, idLookup, Update</p> <p>Description The name of the milestone.</p>
RecurrenceType	<p>Type picklist</p> <p>Properties Create,Update</p> <p>Description The type of recurrence for the milestone.</p>

Usage

Use this object to query and manage the milestone type for CaseMilestone records.

SEE ALSO:

[CaseMilestone](#)
[SlaProcess](#)

Name

Non-queryable object that provides information about foreign key traversals when the foreign key has more than one parent.

This object is used to retrieve information from related records where the related record may be from more than one object type (a polymorphic foreign key). For example, the owner of a case can be either a user or a group (queue). This object allows retrieval of the owner name, whether the owner is a user or a group (queue). You can use a describe call to access the information about parents for an object, or you can use the `who`, `what`, or `owner` fields (depending on the object) in SOQL queries. This object cannot be directly accessed.

Supported Calls

`describeSObjects()`

Fields

Field	Details
Alias	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user alias. This field contains a value only if the related record is a user.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The email address of the user. This field contains a value only if the related record is a user.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name of the user, contact, or lead.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description Indicates whether the related record is an active user (<code>true</code>) or not (<code>false</code>). This field contains a value only if the related record is a user.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The last name of the user, contact, or lead.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
MiddleName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The middle name of the user, contact, or lead.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the parent of the object queried. If the parent is a user, contact, or lead, the value is a concatenation of the <code>FirstName</code>, <code>MiddleName</code>, <code>LastName</code>, and <code>Suffix</code> fields of the related record.</p>

Field	Details
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The phone number of the user. This field contains a value only if the related record is a user.</p>
Profile	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description The Profile of the user. Only populated if the related record is a user.</p>
ProfileId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user's Profile. Only populated if the related record is a user.</p>
Suffix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name suffix of the user, contact, or lead.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The title of the user, for example CFO or CEO.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description A list of the types of sObjects that can be an owner of this object. You can use this field to filter on a type of owner, for example, return only the leads owned by a user.</p>
Username	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Contains the name that a user enters to log into the API or the user interface. The value for this field is in the form of an email address, and is only populated if the related record is a user.</p>
UserRole	<p>Type picklist</p> <p>Properties Filter, Nillable</p> <p>Description Name of the Role played by the user. Only populated for user rows.</p>
UserRoleId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user role associated with this object.</p>

Usage

To query on relationships where the parent may be more than one type of object, use `who`, `what`, or `owner` relationship fields.

SEE ALSO:

[Object Basics](#)

NamedCredential

Represents a named credential, which specifies the URL of a callout endpoint and its required authentication parameters in one definition. A named credential can be specified as an endpoint to simplify the setup of authenticated callouts. This object is available in API version 33.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
<code>CalloutOptionsAllowMergeFieldsInBody</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description For Apex callouts, indicates whether the code can use merge fields to populate HTTP request bodies with org data. This field is available in API version 35.0 and later.</p>
<code>CalloutOptionsAllowMergeFieldsInHeader</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description For Apex callouts, indicates whether the code can use merge fields to populate HTTP headers with org data. This field is available in API version 35.0 and later.</p>
<code>CalloutOptionsGenerateAuthorizationHeader</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates whether Salesforce automatically generates a standard authorization header for each callout to the named credential–defined endpoint. This field is available in API version 35.0 and later.</p>
<code>DeveloperName</code>	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not</p>

Field Name	Details
Endpoint	<p>include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
Language	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description The root URL of the endpoint.</p>
MasterLabel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of the <code>MasterLabel</code>.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The master label for the named credential. This display value is the internal label that doesn't get translated.</p>
PrincipalType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p>

Field Name	Details
	<p>Properties Filter, Group, Restricted picklist, Sort</p>
	<p>Description Tracks users who are accessing the external system. Anonymous implies that a user identity isn't specified for external system access. Named Principal uses one user identity for all users to access the external system.</p>

Usage

Use the NamedCredential object to query named credentials in your organization.

SEE ALSO:

[ExternalDataUserAuth](#)
[ExternalDataSource](#)

NavigationLinkSet

Represents the navigation menu in a community. A navigation menu consists of items that users can click to go to other parts of the community. This object is available in API version 36.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Navigation menus are available only in communities created using the Customer Service (Napili) template.

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two</p>

Field Name	Details
	consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.
Language	Type picklist Properties Filter, Group, Restricted picklist, Sort Description Language for the navigation menu.
MasterLabel	Type string Properties Filter, Group, Sort Description Label for the navigation menu.
NetworkId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the community.

NavigationMenuItem

Represents a single menu item in a NavigationLinkSet. Use this object to create, delete, or update menu items in your community's navigation menu. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Navigation menus are available only in communities created using the Customer Service (Napili) template.

Fields

Field Name	Details
AccessRestriction	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Determines if the menu item is available to guest users who aren't required to log in to the community.</p>
DefaultListViewId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description If the value of the <code>Type</code> field is <code>SalesforceObject</code>, the value is the ID of the default list view for the object.</p>
Label	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text that appears in the navigation menu for this item.</p>
NavigationLinkSetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The navigation menu that this item is included in.</p>
Position	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The location of the menu item in the navigation menu.</p>
Target	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description If <code>Type</code> is ExternalLink or InternalLink, the target is the URL that the link points to. For ExternalLink, your entry looks like this: <code>http://www.salesforce.com</code>. For InternalLink, use a relative URL, such as <code>/contactsupport</code>.</p>
TargetPrefs	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description If <code>Type</code> is ExternalLink, determines whether a navigation menu item opens in the same tab.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of navigation menu item. The available values are:</p> <ul style="list-style-type: none"> • SalesforceObject—Available objects include accounts, cases, contacts, and custom objects. • ExternalLink—Links to a URL outside of your community. For example, <code>http://www.salesforce.com</code>. • InternalLink—Links to a relative URL inside your community. For example, <code>/contactsupport</code>. • NavigationalTopic—A drop-down list with links to the navigational topics in your community.

Usage

You can add up to 20 navigation menu items. You can translate navigation menu items using the Translation Workbench.

NavigationMenuItemLocalization

Represents the translated value of a navigation menu item in a community. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Navigation menus are available only in communities created using the Customer Service (Napili) template.

Fields

Field Name	Details
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The language of the translated navigation menu item. The picklist contains the following supported languages:</p> <ul style="list-style-type: none"> • Chinese (Simplified): zh_CN • Chinese (Traditional): zh_TW • Danish: da • Dutch: nl_NL • English: en_US • Finnish: fi • French: fr • German: de • Italian: it • Japanese: ja • Korean: ko • Norwegian: no • Portuguese (Brazil): pt_BR • Russian: ru • Spanish: es • Spanish (Mexico): es_MX • Swedish: sv • Thai: th
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	<p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the navigation menu item that this translated value applies to.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The translated text for the navigation menu item. Label is Translation Text.</p>

Network

Represents a community, which is a customizable public or private space where employees, end-customers, and partners can collaborate on best practices and business processes. Communities give you the opportunity to share information, records, and files with coworkers and related external stakeholders all in one place. This object is available in API version 26.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve(), update()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
AllowedExtensions	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description Specifies the types of files allowed in your community. This whitelist of file types lets you control what your community members upload and also prevents spammers from polluting your community with inappropriate files. Available in API version 36.0 and later. Separate file types with a comma (for example: <i>jpg, docx, txt</i>). You can enter lowercase and uppercase letters. You can enter up to 1,000 characters. To allow all file types, leave this field empty.</p>
CaseCommentEmailTemplateId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the email template used when submitting a comment on a case. This field is available in API version 28.0 and later.</p>
ChangePasswordEmailTemplateId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the email template used when notifying a user that their password has been reset.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Description of the community.</p>

Field Name	Details
EmailFooterLogoId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the Document object that displays as an image in the footer of community Chatter emails.</p>
EmailFooterText	<p>Type string</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description Text that displays in the footer of community Chatter emails.</p>
EmailSenderAddress	<p>Type email</p> <p>Properties Filter, Group, Sort</p> <p>Description Read only. Email address from which community emails are sent.</p> <p> Note: To change the <code>EmailSenderAddress</code> value, you must first specify <code>NewSenderAddress</code>, which triggers the sending of an address change verification email. After you complete the address verification process, <code>EmailSenderAddress</code> changes to the specified <code>NewSenderAddress</code>.</p>
EmailSenderName	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description Name from which community emails are sent.</p>
FirstActivationDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The date the community was first activated. This field is available in API version 34.0 and later. If the community was activated or inactive prior to the release of API version 34.0, this field returns the date the community was first created.</p>

Field Name	Details
ForgotPasswordEmailTemplateId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the email template used when a user forgets their password.</p>
MaxFileSizeKb	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Specifies the maximum file size (in KBs) that members can upload in your community. Available in API version 36.0 and later. Enter a number between 3072 KB and your org's maximum file size. To use the default limit of 2 GB, leave this field empty.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The name of the community.</p>
NewSenderId	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Email address that has been entered as the new value for <code>EmailSenderId</code> but has not been verified yet. After a user has requested to change the sender email address and has successfully responded to the verification email, the <code>NewSenderId</code> value overwrites the value in <code>EmailSenderId</code>. This becomes the email address from which community emails are sent.</p> <p> Note:</p> <ul style="list-style-type: none"> If verification is pending for a new email address and you set <code>NewSenderId</code> to null, this cancels the verification request. <code>NewSenderId</code> is automatically set to null after <code>EmailSenderId</code> has been set to the new verified address.

Field Name	Details
OptionsAllowInternalUserLogin	<ul style="list-style-type: none"> If verification is pending for a new email address and you specify a different new address for this field, only the latest value is retained and used for verification.
OptionsAllowMembersToFlag	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Specifies whether internal users can log in with their internal credentials on the community login page. This field is available in API version 37.0 and later.</p> <p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Determines whether users can flag posts, comments, or files as inappropriate in the community. This field is available in API version 29.0 and later. The ability to flag files is available in version 30.0 and later.</p>
OptionsGuestChatterEnabled	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Specifies whether guest users can access public Chatter groups in the community without logging in.</p>
OptionsInvitationsEnabled	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Determines whether users can invite others to the community.</p>
OptionsKnowledgeableEnabled	<p>Type boolean</p> <p>Properties Filter</p>

Field Name	Details
	Description
	Determines whether users can see knowledgeable people for topics and endorse people for topics.
OptionsNicknameDisplayEnabled	Type
	boolean
	Properties
	Filter, Update
	Description
	Determines whether user nicknames display instead of their first and last names in most places in the community.
	A few restrictions to keep in mind about nickname display:
	<ul style="list-style-type: none"> • Private messages display full names. You can choose to turn off private messages to avoid this. • Records and user lookups on records show full names. Keep in mind, though, that you can control record and user visibility with sharing rules. • Mobile notifications in Salesforce1 show full names. You can turn off mobile notifications in Salesforce1 to avoid this. • Searches by first, last, and full names aren't restricted and return matches, but the search results display only nicknames. The auto-complete recommendations in global search and the recent items list show any first, last, and full names that the user has already searched by or has accessed via a record or another location.
OptionsPrivateMessagesEnabled	Type
	boolean
	Properties
	Filter
	Description
	Determines whether users can send and receive Chatter messages in the community.
OptionsReputationEnabled	Type
	boolean
	Properties
	Filter, Update
	Description
	Determines if reputation is calculated and displayed for community members. This field is available in API version 31.0 and later.
OptionsSelfRegistrationEnabled	Type
	boolean

Field Name	Details
	Properties Filter, Update
	Description Determines whether external users can self-register to join the community. External users are users with Community, Customer Portal, or partner portal licenses. If true, displays a Not a member? link on the login page that points to the default self-registration page. This field is available in API version 28.0 and later.
OptionsSendWelcomeEmail	Type boolean
	Properties Filter, Update
	Description Determines whether a welcome email is sent when a new user is added to the community.
OptionsShowAllNetworkSettings	Type boolean
	Properties Filter, Update
	Description Determines whether settings in Community Management that were hidden based on how you set up your community are visible or remain hidden. This field is available in API version 33.0 and later.
OptionsSiteAsContainerEnabled	Type boolean
	Properties Filter, Update
	Description Determines whether the community uses Site.com pages instead of tabs. This field is available in API version 29.0 and later.
SelfRegProfileId	Type reference
	Properties Filter, Group, Nillable, Sort, Update
	Description ID of the profile assigned to users who self register. Only applies if self registration is enabled for the community. This field is available in API version 29.0 and later.

Field Name	Details
Status	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Status of the community. Available values are:</p> <ul style="list-style-type: none"> • Live—The community is online and members can access it. Label is Published. • DownForMaintenance—The community was previously published, but was taken offline. Members with “Create and Set Up Communities” can still access the setup for offline communities regardless of profile or membership. Members are not able to access offline communities, but they still appear in the user interface drop-down as <code>CommunityName (Offline)</code>. Label is Offline. • UnderConstruction—The community has not yet been published. Users with “Create and Set Up Communities” can access communities in this status if their profile is associated with the community. <p>Once a community is published, it can never be in this status again. Label is Preview.</p>
UrlPathPrefix	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The <code>UrlPathPrefix</code> is a unique string at the end of the URL for this community. For example, in the community URL <code>UniversalTelco.force.com/customers</code>, <code>customers</code> is the <code>UrlPathPrefix</code>.</p>
WelcomeEmailTemplateId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the email template used when sending welcome emails to new community members.</p>

Usage

Use this object to find, view, and update communities in your organization. If you have “Modify All Data,” “View All Data,” or “Create and Set Up Communities,” you can view all communities in the organization. Users without these permissions only see the `Preview` or

Published communities that they're members of. If you have "Create and Set Up Communities," you can customize community settings.

NetworkActivityAudit

Represents an audit trail of moderation actions in Communities. This object is available in API version 30.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
Action	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort, Update</p> <p>Description The moderation action a member performed on a post, comment, or file in a community. Values are:</p> <ul style="list-style-type: none">Flagged as Inappropriate—A community member flagged a post, comment, or file as inappropriate.Flagged as Spam - A community member flagged a post, comment, or file as spam.Unflagged—A community member removed the flag from a post, comment, or file.RemovedFlags—A moderator removed all flags from a post, comment, or file.DeletedFlaggedItem—A moderator deleted a flagged post, comment, or file.ModerationRuleFlag—A moderation rule flagged member-generated content.ModerationRuleBlock—A moderation rule blocked member-generated content.ModerationRuleReplace—A moderation rule replaced member-generated content.

Field Name	Details
	<ul style="list-style-type: none"> ModerationRuleReview—A moderation rule sent member-generated content to be reviewed and approved by a moderator. ModerationRuleFreeze—A moderation rule froze a member because they created content too frequently within a specific time frame. ModerationRuleNotify—A moderation rule notified moderators because a member created content too frequently within a specific time frame.
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Notes entered by the user. If the entity being tracked is a file, records the version number of the file when it was flagged.</p>
EntityCreatedBy	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description ID of the user that created the entity being tracked.</p>
EntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID of the entity that is being tracked. The following entities are tracked: ChatterMessage, ContentDocument, ContentVersion, FeedComment, and FeedItem.</p>
EntityType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The key prefix of the entity being tracked.</p>
Name	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the item being tracked.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the network (community) where the moderation action was performed.</p>
ParentEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the parent of the entity on which an action was performed. The following entities are tracked: Collaboration Group and User.</p>
ParentEntityType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The key prefix of the parent of the entity being audited.</p>

Usage

Use this object to view an audit trail of moderation activity for your communities. You must have the “Modify All Data” permission to access this object.

Users with “Moderate Communities Feeds”, “Moderate Communities Files” or “View All Data” can view the audit trail using reports in the Salesforce user interface.

NetworkMember

Represents a member of a community. Members can be either users in your company or external users with portal profiles. This object is available in API version 26.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve(), update()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
DefaultGroupNotificationFrequency	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. The default frequency for sending the member's group email notifications when the member joins groups in the community. The valid values are:</p> <ul style="list-style-type: none"> • P—Email on every post • D—Daily digests • W—Weekly digests • N—Never <p>The default value is W. However, this field is not currently enabled. These values are reserved for future use.</p>
DigestFrequency	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. The frequency for sending the member's personal email digest for the community. The valid values are:</p> <ul style="list-style-type: none"> • D—Daily • W—Weekly • N—Never <p>The default value is D. However, daily and weekly personal digests aren't currently available in communities. These values are reserved for future use.</p>
LastChatterActivityDate	<p>Type date</p>

Field Name	Details
MemberId	Properties Filter, Group, Nillable, Sort
	Description The last time the member posted or commented in the community.
NetworkId	Type reference
	Properties Filter, Group, Sort
	Description The ID of a person who is a member of a community.
PreferencesDisableAllFeedsEmail	Type reference
	Properties Filter, Group, Sort
	Description The ID of the community that the member is part of.
PreferencesDisableBookmarkEmail	Type boolean
	Properties Filter, Update
	Description When <code>false</code> , the member can automatically receive email for updates in the community, based on the types of feed emails and digests the member has enabled.
PreferencesDisableChangeCommentEmail	Type boolean
	Properties Filter, Update
	Description When <code>false</code> , the member automatically receives email every time someone comments on a feed item after the member has bookmarked it.
PreferencesDisableCommentEmail	Type boolean
	Properties Filter, Update

Field Name	Details
PreferencesDisableEndorsementEmail	<p>Description When <code>false</code>, the member automatically receives email every time someone comments on a change the member has made, such as an update to their profile.</p>
PreferencesDisableFollowersEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone endorses them for a topic.</p>
DisableItemFlaggedEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone in the community starts following the member.</p>
PreferencesDisableLaterCommentEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time a community member flags a post or comment. This setting only applies for community moderators (with the "Moderate Communities Feeds" permission) and group owners or managers.</p> <p>This field is available in API version 29.0 and later.</p>

Field Name	Details
PreferencesDisableLikeEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone comments on a feed item after the member has liked the feed item.</p>
PreferencesDisableMentionsPostEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time the member is mentioned in posts.</p>
PreferencesDisableMessageEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time the member is sent a Chatter message.</p>
PreferencesDisableProfilePostEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone posts to the member's profile.</p>
PreferencesDisableSharePostEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time the member's post is shared.</p>

Field Name	Details
PreferencesDisCommentAfterLikeEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone comments on a post the member has liked.</p>
PreferencesDisMentionsCommentEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time the member is mentioned in comments.</p>
PreferencesDisProfPostCommentEmail	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone comments on posts on the member's profile.</p>
ReputationPoints	<p>Type double</p> <p>Properties Filter, Sort, Update</p> <p>Description The number of reputation points the user has accumulated by performing actions in the community.</p>

Usage

Use this object to query members of a certain community and to update their email notification settings. If you have "Modify All Data," "View All Data," or "Create and Set Up Communities," you can view all members of any community, regardless of your own membership. If you have "Modify All Data" or "Create and Set Up Communities," you can also update any member's email settings. Users without these permissions can update their own email settings and can see members of the communities that they're also members of.

 **Tip:** You can directly update reputation points for a community member via the Salesforce API. You can also use Apex triggers to send custom notifications based on changes to reputation points.

NetworkMemberGroup

Represents a group of members in a community. Members can be either users in your internal organization or external users assigned portal profiles. An administrator adds members to a community by adding a profile or a permission set, and any user with the profile or permission set becomes a member of the community. This object is available in API version 26.0 and later.

 **Note:** If a Chatter customer (from a customer group) is assigned a permission set that is also associated with a community, the Chatter customer won't be added to the community.

Prior to API version 27.0, this object was called NetworkProfile.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

 **Note:** The `upsert()` call is not supported for this object.

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
AssignmentStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status of a profile or permission set within a community. Values are:</p> <ul style="list-style-type: none"> Added—Users with this profile or permission set are members of the community. Waiting for Add—The profile or permission set was added to the community, but the async process hasn't completed yet. After the process is complete, the status is updated to Added. Waiting for Remove—Use this status to remove all the community members belonging to a profile or permission set and remove a profile or permission set from a community.
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description The ID of the community that this group of members is associated with.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the profile or permission set associated with the community.</p>

Usage

Use this object to view the profiles or permission sets associated with a particular community. Profiles and permission sets are added and removed asynchronously, so you can also check the status of a profile or permission set that was updated in a community.

If you have “Modify All Data,” “View All Data,” or “Create and Set Up Communities,” you can view all profiles or permission sets for any community in the organization, regardless of your membership. If you have “Modify All Data” or “Create and Set Up Communities,” you can also add profiles or permission sets. Users without these permissions can only find profiles and permission sets for communities that they’re members of.

Sample Code

```
// Create a new NetworkMemberGroup with a profile as the ParentId
NetworkMemberGroup nmgInsert = new NetworkMemberGroup();
nmg.setNetworkId("0DBD0000000029o");
nmg.setParentId("00eD0000000z1Ww");
SaveResult[] results = connection.create(new SObject[] { nmgInsert });

// Update an existing NetworkMemberGroup to be removed from the Network
NetworkMemberGroup nmgUpdate = new NetworkMemberGroup();
nmg.setId("0LD00000003enOAA");
nmg.setAssignmentStatus("WaitingForRemove");
SaveResult[] results = connection.update(new SObject[] { nmgUpdate });
```

NetworkModeration

Represents a flag on an item in a community. This object is available in API version 30.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization.

Fields

Field Name	Details
EntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the post, comment, or file that was flagged.</p>
ModerationType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort</p> <p>Description Determines the type of flag applied to an item. Values are:</p> <ul style="list-style-type: none"> • FlagAsInappropriate • FlagAsSpam
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the community in which the item was flagged.</p>
Visibility	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Nillable, Sort</p> <p>Description Determines visibility of a flagged item. Values are:</p> <ul style="list-style-type: none"> • SelfAndModerators—The user who flagged the item and any moderators can see the flagged item. This is the default value. • ModeratorsOnly—Only moderators can see the flagged item. If ModeratorsOnly is selected, only moderators can set flags using the API.

Usage

Use this object to view the items flagged for moderation within a community. Additionally, users with “Moderate Feeds” and “Modify All Data” can remove flags.

Flags on items are created either when a member manually flags an item in a community (if flagging is enabled for that community), or when a trigger automatically flags an item because the item met the trigger criteria.

NetworkPageOverride

Represents information about custom pages used to override the default pages in communities. You can create Community Builder (Site.com Studio) or Visualforce pages and override the default pages in a community. Using custom pages allows you to create a more personalized experience for your users. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

- Only users with the “Create and Setup Networks” permission can update this object.
- You can’t override the Change Password Page with a page created using Community Builder (Site.com Studio). You can only override it with a Visualforce page.

Fields

Field Name	Details
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the community where a custom page is used to override a default page.</p>
OverrideSetting	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of the page used to override a default page in the community. <code>OverrideSetting</code> can take the following values:</p> <ul style="list-style-type: none">Standard—The standard page that comes by default with the community.Designer—A custom page created using Community Builder (Site.com Studio).

Field Name	Details
	<ul style="list-style-type: none"> Visualforce—A custom page created using Visualforce.
OverrideType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The name of the default page in the community that you want to override with a custom page. <code>OverrideType</code> can take the following values:</p> <ul style="list-style-type: none"> LoginRequired ChangePassword ForgotPassword SelfReg Home

NetworkSelfRegistration

Represents the account that self-registering community members are associated with by default. Self-registering users in a community are required to be associated with an account, which the administrator must specify while setting up self-registration for the community. If an account isn't specified, Salesforce creates person accounts (when enabled) for self-registering users. This object is available in API version 34.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the account that self-registering users in the community are associated with.</p>
NetworkId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the community.</p> <p> Note: You can use only one account per community to assign self-registering users.</p>

NewsFeed

Represents a single feed item on a user's home tab. A Chatter feed shows recent changes to records that the user is following.

NewsFeed is available in API version 18.0 through API version 26.0. In API version 27.0 and later, NewsFeed is no longer available in the SOAP API. Use the Chatter REST API to access NewsFeed.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of NewsFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your</p>

Field	Details
	<p>comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection makes a change to a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator while the <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection that made the change. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>

Field	Details
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the newsfeed record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The title of the feed item. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL and this field is the link name.</p>
Type	<p>Type</p> <p>picklist</p>

Field	Details
	<p>Properties</p> <p>Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description</p> <p>The type of feed item:</p> <ul style="list-style-type: none">• ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RipplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• CaseCommentPost—generated event when a user adds a case comment for a case object• EmailMessageEvent—generated event when an email related to a case object is sent or received

Field	Details
	<ul style="list-style-type: none"> • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. This event is also generated by CTI calls. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, you must specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to query and retrieve record changes tracked in a news feed. Note the following when working with news feeds:

- This object is read only in the API.
- Users can only query their own news feed.
- Queries retrieve feed items that include mentions from other users.
- Use this object to query and retrieve lead feed items that were associated with a converted lead record.

Users that do not have the “View All Data” permission have the following limitations when querying records:

- Must specify a `LIMIT` clause and the limit must be less than or equal to 1000.
- May include a `WHERE` clause that references NewsFeed fields, but may not include references to fields in related objects. For example, you can filter by `CreatedDate` or `ParentId`, but not by `Parent.Name`.
- May include an `ORDER BY` clause that references NewsFeed fields, but may not include references to fields in related objects. For example, you can `ORDER BY CreatedDate` or `ParentId`, but not by `Parent.Name`.

 **Tip:** To query for the most recent feed items, you should ORDER BY CreatedDate DESC, Id DESC.

Note the following SOQL restrictions.

- No SOQL limit if logged-in user has "View All Data" permission. If not, specify a LIMIT clause of 1,000 records or fewer.
- SOQL ORDER BY on fields using relationships is not available. Use ORDER BY on fields on the root object in the SOQL query.

SEE ALSO:

[EntitySubscription](#)

[FeedComment](#)

[FeedTrackedChange](#)

Note

Represents a note, which is text associated with a custom object or a standard object, such as a Contact, Contract, or Opportunity.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()`

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Body of the note. Limited to 32 KB.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrivate	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description If true, only the note owner or a user with the "Modify All Data" permission can view the note or query it via the API. Note that if a user who does not have the "Modify All Data" permission sets this field to true on a note that they do not own, then they can no longer query, delete, or update the note. Label is Private.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns the note.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the object associated with the note.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Title of the note.</p>

Usage

Use this object to manage notes for an object.

SEE ALSO:

[Object Basics](#)

OAuthToken

Represents an OAuth access token for connected app authentication and can be used to create a user interface for token management. This object is available in API version 32.0 and later.

A connected app integrates an application with Salesforce using APIs. Connected apps use standard SAML and OAuth protocols to authenticate, provide single sign-on, and provide tokens for use with Salesforce APIs. In addition to standard OAuth capabilities, connected

apps allow Salesforce admins to set various security policies and have explicit control over who can use the corresponding apps. Each time that a user grants access to an application, the application obtains a new access token.

Supported Calls

`describeSObjects(), query()`

Special Access Rules

If you are a user with the “Manage Users” permission, you see all tokens for all users in the organization. Otherwise, you see only your own tokens.

Fields

Field Name	Details
AccessToken	Type string Properties Filter, Group, Nillable, Sort Description The refresh token for authorization.
AppMenuItemId	Type reference Properties Filter, Group, Nillable, Sort Description The unique ID for the App Picker menu item that’s associated with this OAuth token.
AppName	Type string Properties Filter, Group, Nillable, Sort Description The label for the connected app that’s associated with this OAuth token.
DeleteToken	Type string Properties Filter, Group, Nillable, Sort

Field Name	Details
	Description
	A token that can be used at the revoke OAuth token endpoint to remove this token.
Id	Type
	ID
	Properties
	Defaulted on create, Filter, Group, idLookup, Sort
	Description
	Reserved for future use. Currently, the value is always null.
LastUsedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The most recent date when the OAuth token was used.
RequestToken	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The authorization code that was used to request the corresponding AccessToken. With this authorization code, you can revoke the corresponding AccessToken by passing the DeleteToken.
UseCount	Type
	int
	Properties
	Filter, Group, Nillable, Sort
	Description
	How often the token has been used.
UserId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	The owner of the token.

Usage

To delete an AccessToken, send a request to the revoke OAuth token endpoint with the DeleteToken as the parameter. For example, this URL `https://login.salesforce.com/services/oauth2/revoke?token=(the Delete Token)` causes the deletion of the token.

In API version 34.0 and later, this object was enhanced to help manage high instance counts. A `query()` call returns up to 500 rows. A `queryMore()` call returns 500 more, up to 2500 total. No more records are returned after 2500. To make sure you don't miss any records, issue a `COUNT()` query in a `SELECT` clause for OauthToken. This gives you the total number of records. If there are more than 2500 records, divide your query by filtering on fields, like `UserId`, to return subsets of less than 2500 records.

NoteAndAttachment

This read-only object contains all notes and attachments associated with an object.

Supported Calls

`describeSObjects()`

Fields

Field	Details
<code>IsDeleted</code>	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
<code>IsNote</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the object contains a note (<code>true</code>) or an attachment (<code>false</code>).
<code>IsPrivate</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description If <code>true</code> , only the note owner or a user with the "Modify All Data" permission can view the note or query it via the API. Note that if a regular user who does not have "Modify All Data"

Field	Details
	permission sets this field to <code>true</code> on a note that they do not own, then they can no longer query, delete, or update that note. Label is Private .
OwnerId	Type reference Properties Filter, Group, Sort Description ID of the user who owns the note and attachment.
ParentId	Type reference Properties Filter, Group, Sort Description ID of the parent object.
Title	Type string Properties Filter, Nillable, Group, Sort Description Title of the note.

Usage

Use this object to list all notes and attachments for an object.

To retrieve notes and attachments, issue a describe call on an object, which returns a query result for each activity since the record was created. You can't directly query this object.

SEE ALSO:

[Note](#)
[Attachment](#)

NoteTag

Associates a word or short phrase with a Note.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

NoteTag stores the relationship between its parent TagDefinition and the Note being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

ObjectPermissions

Represents the enabled object permissions for the parent PermissionSet. This object is available in API version 24.0 and later.

To grant a user access to an object, associate an ObjectPermissions record with a PermissionSet that's assigned to a user. ObjectPermissions records are only supported in PermissionSet, not in Profile.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The Id of this object's parent PermissionSet.</p>
PermissionsCreate	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description If true, users assigned to the parent PermissionSet can create records for this object. Requires PermissionsRead for the same object to be true.</p>
PermissionsDelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description If true, users assigned to the parent PermissionSet can delete records for this object. Requires PermissionsRead and PermissionsEdit for the same object to be true.</p>
PermissionsEdit	<p>Type boolean</p>

Field Name	Details
	Properties
	Create, Filter, Update
	Description
	If <code>true</code> , users assigned to the parent PermissionSet can edit records for this object. Requires <code>PermissionsRead</code> for the same object to be <code>true</code> .
PermissionsModifyAllRecords	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	If <code>true</code> , users assigned to the parent PermissionSet can edit all records for this object, regardless of sharing settings. Requires <code>PermissionsRead</code> , <code>PermissionsDelete</code> , <code>PermissionsEdit</code> , and <code>PermissionsViewAllRecords</code> for the same object to be <code>true</code> .
PermissionsRead	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	If <code>true</code> , users assigned to the parent PermissionSet can view records for this object.
PermissionsViewAllRecords	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	If <code>true</code> , users assigned to the parent PermissionSet can view all records for this object, regardless of sharing settings. Requires <code>PermissionsRead</code> for the same object to be <code>true</code> .
SObjectType	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort
	Description
	The object's API name. For example, <code>Merchandise__c</code> .

Permission Dependencies

Some user permissions have dependencies on object permissions. For example, if a permission set has the “Transfer Leads” permission, it also has “Read” and “Create” on the leads object.

You can query from ObjectPermissions up to the parent PermissionSet object. For example:

```
SELECT Parent.Name, Parent.PermissionsTransferAnyLead, PermissionsRead, PermissionsCreate
FROM ObjectPermissions
WHERE SObjectType = 'Lead'
```

Determining Object Access with “Modify All Data”

When using SOQL to query object permissions, be aware that some object permissions are enabled because a user permission requires them.

The exception to this rule is when “Modify All Data” is enabled. While it enables all object permissions, it doesn’t physically store any object permission records in the database. As a result, unlike object permissions that are required by a user permission—such as “View All Data” or “Import Leads”—the query still returns permission sets with “Modify All Data,” but the object permission record will contain an invalid ID that begins with “000”. This ID indicates that the object has full access due to “Modify All Data” and the object permission record can’t be updated or deleted. To remove full access from these objects, disable “Modify All Data” and then delete the resulting object permission record. This ensures that when using SOQL to find all the objects that have full access, it returns all objects that have this access regardless of whether it’s due to “Modify All Data” or because an administrator set full access.

For example, the following will return all permission sets that have “Read” on the Merchandise__c object, regardless of whether it’s explicitly defined on the object or implicitly defined through “Modify All Data.”

```
SELECT Id, Parent.label, SObjectType, PermissionsRead,
       Parent.PermissionsModifyAllData, ParentId
FROM ObjectPermissions
WHERE PermissionsRead = true and SObjectType = 'Merchandise__c'
```

Nesting Object Permissions

You can nest ObjectPermissions in a PermissionSet query. For example, the following returns any permission sets where “Transfer Leads” is true. Additionally, the result set will include the “Read” object permission on leads. This is done by nesting the SOQL with an object permission query using the relationship name for object permissions: ObjectPerms.

```
SELECT Id, Name, PermissionsTransferAnyLead,
       (SELECT Id, PermissionsRead from ObjectPerms where SObjectType='Lead')
FROM PermissionSet
WHERE PermissionsTransferAnyLead = true
```

As a result, it’s possible to traverse the relationship between the PermissionSet and any child related objects (in this case, ObjectPermissions). You can do this from the PermissionSet object by using the child relationship (ObjectPerms, FieldPerms, and so on) or from the child object by referencing the PermissionSet with **Parent.permission_set_attribute**.

It’s important to consider when to use a conditional WHERE statement to restrict the result set. To query based on an attribute on the permission set object, nest the SOQL with the child relationship. However, to query based on an attribute on the child object, you must reference the permission set parent attribute in your query.

The following two queries return the same columns with different results, based on whether you use the child relationship or parent notation.

```
SELECT Id, Name, PermissionsModifyAllData,
(SELECT Id, SObjectType, PermissionsRead from Objectperms)
FROM PermissionSet
WHERE PermissionsModifyAllData=true
```

versus:

```
SELECT Id, SObjectType, PermissionsRead, Parent.Id, Parent.Name,
Parent.PermissionsModifyAllData
FROM ObjectPermissions
WHERE SObjectType='Merchandise__c'
```

SEE ALSO:

[PermissionSet](#)
[FieldPermissions](#)

ObjectTerritory2AssignmentRule

Represents a territory assignment rule that's associated with an object, such as Account. ObjectTerritory2AssignmentRuleItem can only be created or deleted if the BooleanFilter field on its corresponding ObjectTerritory2AssignmentRule is null. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories, assignment rules, assigned records, and assigned users. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
BooleanFilter	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Represents advanced filter conditions that were specified for the rule in the online application. For example, "(1 AND 2) OR 3."</p>

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. The field label in the user interface is Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the rule is active (true) or inactive (false). Via the API, active rules run automatically when object records are created and edited. The exception is when the value of the IsExcludedFromRealign field on an object record is true, which prevents record assignment rules from evaluating that record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the master label in the user interface.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The user interface label for the territory type.</p>
ObjectType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description The object that the rule is defined for. For API version 31, Account only.</p>
Territory2ModelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the territory model.</p>

ObjectTerritory2AssignmentRuleItem

A single row of selection criteria for an ObjectTerritory2AssignmentRule object. ObjectTerritory2AssignmentRuleItem can only be created or deleted if the BooleanFilter field on its corresponding ObjectTerritory2AssignmentRule object is a null value. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

```
create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()
```

Special Access Rules

If a territory model is in Active state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your organization's sharing settings. Users cannot view territory models in other states (such as Planning or Archived).

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The standard or custom object field that the rule item will operate on.</p>
Operation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description The criterion to apply for the rule item. For example: <i>equals</i> or <i>starts with</i>.</p>
RuleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the associated ObjectTerritory2AssignmentRule.</p>
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The order in which this row is evaluated in relation to other ObjectTerritoryAssignmentRuleItem objects for the given ObjectTerritoryAssignmentRule.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The field value or values to evaluate. For example: if the field is <code>Billing ZIP/Postal Code</code>, a value might be <code>94105..</code></p>

ObjectTerritory2Association

Represents an association (by assignment) between a territory and an object record, such as an account. Available only if Enterprise Territory Management has been enabled for your Salesforce org.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your org's sharing settings. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
AssociationCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The means by which the record was associated with the territory: by rules or manually. User interface field label is <code>Method</code>. Manual cause can be set only by a user or via the API. Rules cause is used by the rules engine when the object is assigned.</p>
ObjectId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the object. For API version 30.0 and later, Account only.</p>
SObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of the record. For API version 30.0 and later, Account only.</p>
Territory2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the territory that the record is assigned to.</p>

OpenActivity

This read-only object is displayed in a related list of open activities—future events and open tasks—related to an object. It includes activities for all contacts related to the object. OpenActivity fields for phone calls are only available if your organization uses Salesforce CRM Call Center.

Supported Calls

```
describeSObjects()
```

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the related account, which is determined as follows:</p> <ul style="list-style-type: none">• The account associated with the <code>WhatId</code>, if it exists; or• The account associated with the <code>WhoId</code>, if it exists; otherwise• <code>null</code> <p>For information on IDs, see ID Field Type.</p>
ActivityDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates one of the following:</p> <ul style="list-style-type: none">• The due date of a task• The date of an event if <code>IsAllDayEvent</code> is set to <code>true</code> <p>This field has a time stamp that is always set to midnight in the Universal Time Coordinated (UTC) time zone. The time stamp doesn't represent the time of the activity; don't attempt to alter it to accommodate time zone differences. Label is <code>Date</code>.</p>
ActivitySubtype	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description Provides standard subtypes to facilitate creating and searching for specific activity subtypes. This field isn't updateable.</p> <p>ActivitySubtype values:</p> <ul style="list-style-type: none"> • Task • Email • Call • Event
ActivityType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents one of the following values: Call, Meeting, or Other. Label is Type.</p>
AlternateDetailId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of a record the activity is related to which contains more details about the activity. For example, an activity can be related to an EmailMessage record.</p>
CallDisposition	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the result of a given call, for example, "we'll call back," or "call unsuccessful." Limit is 255 characters.</p>
CallDurationInSeconds	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Duration of the call in seconds.</p>
CallObject	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of a call center. Limit is 255 characters.</p>
CallType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of call being answered: Inbound, Internal, or Outbound.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the PartnerNetworkConnection that shared this record with your organization. This field is available only if your organization has enabled Salesforce to Salesforce and only in API versions 28.0 and later.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the PartnerNetworkConnection that your organization shared this record with. This field is available only if your organization has enabled Salesforce to Salesforce, and only in API versions 28.0 and later. The value is always <code>null</code>. You can use the <code>PartnerNetworkRecordConnection</code> object to forward records to connections.</p>
Description	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description Contains a description of the event or task. Limit is 32 KB.</p>
DurationInMinutes	<p>Type int</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the duration of the event or task.</p>
EndDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates the end date and time of the event or task. Available in versions 27.0 and later. This field is optional, depending on the following:</p> <ul style="list-style-type: none"> • If <code>IsAllDayEvent</code> is true, you can supply a value for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are <code>null</code>, the duration defaults to one day. • If <code>IsAllDayEvent</code> is false, a value must be supplied for either <code>DurationInMinutes</code> or <code>EndDateTime</code>. Supplying values in both fields is allowed if the values add up to the same amount of time.
IsAllDayEvent	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If the value of this field is set to <code>true</code>, then the activity is an event spanning a full day, and the <code>ActivityDate</code> defines the date of the event. If the value of this field is set to <code>false</code>, then the activity may be an event spanning less than a full day, or it may be a task. Label is <code>All-Day Event</code>.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a task is closed; value is always <code>false</code>). This field is set indirectly by setting <code>Status</code> on the task—each picklist value has a corresponding <code>IsClosed</code> value. Label is <code>Closed</code>.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p>

Field	Details
Description	Indicates whether the activity has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is <code>Deleted</code> .
<code>IsHighPriority</code>	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	Indicates a high-priority task. This field is derived from the <code>Priority</code> field.
<code>IsReminderSet</code>	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	Indicates whether a reminder is set for an activity (<code>true</code>) or not (<code>false</code>).
<code>IsTask</code>	Type boolean
Properties	Defaulted on create, Filter, Group, Sort
Description	If the value of this field is set to <code>true</code> , then the activity is a task; if the value is set to <code>false</code> , then the activity is an event. Label is <code>Task</code> .
<code>IsVisibleInSelfService</code>	Type boolean
Properties	Defaulted on create, Filter
Description	If the value of this field is set to <code>true</code> , then the activity can be viewed in the self-service portal. Label is <code>Visible in Self-Service</code> .
<code>Location</code>	Type string
Properties	Filter, Group, Nillable, Sort
Description	If the activity is an event, then this field represents the location of the event. If the activity is a task, then the value is <code>null</code> .

Field	Details
OwnerId	Type reference
	Properties Filter, Group, Nillable, Sort
	Description Indicates the ID of the user who owns the activity.
PrimaryAccountId	Type reference
	Properties Filter, Group, Nillable, Sort
	Description Contains the AccountId value from the activity record. Available in API versions 30.0 and later to organizations that use Shared Activities.
PrimaryWhoId	Type reference
	Properties Filter, Group, Nillable, Sort
	Description Contains the WhoId value from the activity record. Available in API versions 30.0 and later to organizations that have enabled Shared Activities.
Priority	Type picklist
	Properties Defaulted on create, Filter, Group, Nillable, Sort
	Description Indicates the priority of a task, such as high, normal, or low.
ReminderDateTime	Type dateTime
	Properties Filter, Nillable, Sort
	Description Represents the time at which a reminder is scheduled to fire if IsReminderSet is set to true. If IsReminderSet is set to false, then either the user has deselected the reminder checkbox in the user interface or the reminder has already fired at the time indicated by the value.
Status	Type picklist

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Nullable, Sort</p> <p>Description Indicates the current status of a task, such as in progress or complete. Each predefined status field sets a value for <code>IsClosed</code>. To obtain picklist values, query <code>TaskStatus</code>.</p>
Subject	<p>Type combobox</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Contains the subject of the task or event.</p>
WhatId	<p>Type reference</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The <code>WhatId</code> represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. <code>WhatIds</code> are polymorphic. Polymorphic means a <code>WhatId</code> is equivalent to the ID of a related object. The label is <code>Related To ID</code>.</p>
WhoId	<p>Type reference</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The <code>WhoId</code> represents a human such as a lead or a contact. <code>Whoids</code> are polymorphic. Polymorphic means a <code>WhoId</code> is equivalent to a contact's ID or a lead's ID. The label is <code>Name ID</code>. If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the <code>WhoId</code> field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is <code>Name ID</code>.</p>

Usage

Query activities that are related to an object

1. Optionally, issue a describe call against the object whose activities you want to query, to get a suggestion of the correct SOQL query to use.

2. Issue a SOQL relationship query with a main clause that references the object and an inner clause that references the activity history. For example:

```
SELECT
  (SELECT ActivityDate, Description
   FROM OpenActivities)
  FROM Account
 WHERE Name Like 'XYZ%'
```

The user interface enforces sharing rules, filtering out related-list items that a user doesn't have permission to see.

The following constraints on users who don't have the "View All Data" permission help prevent performance issues.

- In the main clause of the relationship query, you can reference only one record. For example, you can't filter on all records where the account name starts with "A." Instead, you must reference a single account record.

```
SELECT
  (SELECT ActivityDate, Description
   FROM OpenActivities
   ORDER BY ActivityDate ASC NULLS LAST, LastModifiedDate DESC
   LIMIT 500)
  FROM Account
 WHERE Name = 'Acme'
 LIMIT 1
```

- In the inner clause of the query, you can't use WHERE.
- In the inner clause of the query, you must specify a limit of 500 or fewer on the number of rows that are returned in the list.
- In the inner clause of the query, you must sort on ActivityDate in ascending order and LastModifiedDate in descending order. You can optionally display nulls last. For example: ORDER BY ActivityDate ASC NULLS LAST, LastModifiedDate DESC.

SEE ALSO:

[Task](#)

OperatingHours

Represents the hours in which a service territory, service resource, or account is available for field service work. This object is available in API version 38.0 and later.

Supported Calls

`create()` `delete()` `describeLayout()` `describeSObjects()` `getDeleted()` `getUpdated()` `query()` `retrieve()` ~~`search()`~~ `undelete()` `update()` `upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the operating hours. Add any details that aren't included in the name.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the operating hours record was last modified. Its label in the user interface is <code>Last Modified Date</code>.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the operating hours record was last viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the operating hours. For example, <i>Summer Hours</i>, <i>Winter Hours</i>, or <i>Peak Season Hours</i>.</p>
TimeZone	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The time zone which the operating hours fall within.</p>

Usage

By default, only System Administrators can view, create, and assign operating hours.

Service territory members—which are service resources who can work in the territory—automatically use their service territory's operating hours. If a resource needs different operating hours than their territory, create separate operating hours for them from the Operating Hours tab. Then, select the desired hours in the `Operating Hours` lookup field on the service territory member detail page.

To view a service resource's operating hours for a particular territory, navigate to their Service Territories related list and click the Member Number for the territory. This takes you to the service territory member detail page, which lists the member's operating hours and dates during which they belong to the territory.

OperatingHoursFeed

Represents a single feed item on an operating hours record detail page. This object is available in API version 38.0 and later.

An operating hours feed shows changes to tracked fields on an operating hours record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to operating hours in Salesforce.

Supported Calls

`delete()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Operating Hours object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p>

Field Name	Details
	Description
	<p>The number of comments associated with this feed item.</p>
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	<p>ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	<p><code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	Type
	int
	Properties
	Filter, Group, Sort
	Description
	<p>The number of feed likes associated with this feed item.</p>
LinkUrl	Type
	url
	Properties
	Nillable, Sort
	Description
	<p>The URL of a link post.</p>
ParentId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	<p>ID of the operating hours record tracked in the feed. The feed is displayed on the detail page for this record.</p>

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

OperatingHoursHistory

Represents the history of changes made to tracked fields on an operating hours record. This object is available in API version 38.0 and later.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and field tracking for operating hours fields must be configured.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
TimeSlotId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the operating hours record being tracked. The history is displayed on the detail page for this record.</p>

Opportunity

Represents an opportunity, which is a sale or pending deal.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Field	Field Type
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the account associated with this opportunity.</p>
Amount	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Estimated total sale amount. For opportunities with products, the amount is the sum of the related products. Any attempt to update this field, if the record has products, will be ignored. The update call will not be rejected, and other fields will be updated as specified, but the Amount will be unchanged.</p>
CampaignId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of a related Campaign. This field is defined only for those organizations that have the campaign feature Campaigns enabled. The User must have read access rights to the cross-referenced Campaign object in order to create or update that campaign into this field on the opportunity.</p>
CloseDate	<p>Type date</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Date when the opportunity is expected to close.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p>

Field	Field Type
	<p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference Properties Filter, Nillable Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
ContractId	<p>Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the contract that's associated with this opportunity.</p>
CurrencyIsoCode	<p>Type picklist Properties Create, Defaulted on create, Filter, Nillable, Restricted picklist, Update Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. If the organization has multicurrency and a Pricebook2 is specified on the opportunity (i.e., the Pricebook2Id field is not blank), then the currency value of this field must match the currency of the PricebookEntry records that are associated with any opportunity line items it has.</p>
Description	<p>Type textarea Properties Create, Nillable, Update Description Text description of the opportunity. Limit: 32,000 characters.</p>

Field	Field Type
ExpectedRevenue	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only field that is equal to the product of the opportunity <code>Amount</code> field and the <code>Probability</code>. You can't directly set this field, but you can indirectly set it by setting the <code>Amount</code> or <code>Probability</code> fields.</p>
Fiscal	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If fiscal years are not enabled, the name of the fiscal quarter or period in which the opportunity <code>CloseDate</code> falls. Value should be in YYY Q format, for example, '2006 1' for first quarter of 2006.</p>
FiscalQuarter	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the fiscal quarter. Valid values are 1, 2, 3, or 4.</p>
FiscalYear	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the fiscal year, for example, 2006.</p>
ForecastCategory	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Restricted picklist field. It is implied, but not directly controlled, by the <code>StageName</code> field. You can override this field to a different value than is implied by the <code>StageName</code> value. The values of this field are fixed enumerated values. The field labels are localized to the language of the user performing the operation, if localized versions of those labels are available for that language in the user interface.</p>

Field	Field Type
<p>In API version 12.0 and later, the value of this field is automatically set based on the value of the <code>ForecastCategoryName</code> and can't be updated any other way. The field properties Create, Defaulted on create, Nillable, and Update are not available in version 12.0.</p>	
<code>ForecastCategoryName</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available in API version 12.0 and later. The name of the forecast category. It is implied, but not directly controlled, by the <code>StageName</code> field. You can override this field to a different value than is implied by the <code>StageName</code> value.
<code>HasOpenActivity</code>	Type boolean Properties Defaulted on create, Group, Description Indicates whether an opportunity has an open event or task (<code>true</code>) or not (<code>false</code>). Available in API version 35.0 and later.
<code>HasOpportunityLineItem</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Read-only field that indicates whether the opportunity has associated line items. A value of <code>true</code> means that Opportunity line items have been created for the opportunity. An opportunity can have opportunity line items only if the opportunity has a price book. The opportunity line items must correspond to PricebookEntry objects that are listed in the opportunity Pricebook2. However, you can insert opportunity line items on an opportunity that does not have an associated Pricebook2. For the first opportunity line item that you insert on an opportunity without a Pricebook2, the API automatically sets the <code>Pricebook2Id</code> field, if the opportunity line item corresponds to a PricebookEntry in an active Pricebook2 that has a <code>CurrencyIsoCode</code> field that matches the <code>CurrencyIsoCode</code> field of the opportunity. If the Pricebook2 is not active or the <code>CurrencyIsoCode</code> fields do not match, then the API returns an error. You can't update the <code>Pricebook2Id</code> or <code>PricebookId</code> fields if opportunity line items exist on the Opportunity. You must delete the line items before attempting to update the <code>PricebookId</code> field.

Field	Field Type
HasOverdueTask	<p>Type boolean</p> <p>Properties Defaulted on create, Group,</p> <p>Description Indicates whether an opportunity has an overdue task (<code>true</code>) or not (<code>false</code>). Available in API version 35.0 and later.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Directly controlled by <code>StageName</code>. You can query and filter on this field, but you can't directly set it in a create, upsert, or update request. It can only be set via <code>StageName</code>. Label is Closed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsExcludedFromTerritory2Filter	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Used for Filter-Based Opportunity Territory Assignment (Pilot in Spring '15 / API version 33). Indicates whether the opportunity is excluded (<code>True</code>) or included (<code>False</code>) each time the APEX filter is executed.</p>
IsSplit	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only field that indicates whether credit for the opportunity is split between opportunity team members. Label is <code>IsSplit</code>. This field is available in versions 14.0 and later for organizations that enabled Opportunity Splits during the pilot period.</p>

Field	Field Type
	 Warning: This field should not be used. However, it's documented for the benefit of pilot customers who find references to <code>IsSplit</code> in code.
IsWon	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Directly controlled by <code>StageName</code> . You can query and filter on this field, but you can't directly set the value. It can only be set via <code>StageName</code> . Label is Won .
LastActivityDate	Type date Properties Filter, Group, Nillable, Sort Description Value is one of the following, whichever is the most recent: <ul style="list-style-type: none"> • Due date of the most recent event logged against the record. • Due date of the most recently closed task associated with the record.
LastReferencedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
LeadSource	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Source of this opportunity, such as Advertisement or Trade Show.

Field	Field Type
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. A name for this opportunity. Limit: 120 characters.</p>
NextStep	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Description of next task in closing opportunity. Limit: 255 characters.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the User who has been assigned to work this opportunity. If you update this field, the previous owner's access becomes Read Only or the access specified in your organization-wide default for opportunities, whichever is greater. If you have set up opportunity teams in your organization, updating this field has different consequences depending on your version of the API: <ul style="list-style-type: none"> • For API version 12.0 and later, sharing records are kept, as they are for all objects. • For API version before 12.0, sharing records are deleted. • For API version 16.0 and later, users must have the "Transfer Record" permission in order to update (transfer) account ownership using this field. </p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of a related Pricebook2 object. The <code>Pricebook2Id</code> field indicates which Pricebook2 applies to this opportunity. The <code>Pricebook2Id</code> field is defined only for those organizations that have products enabled as a feature. You can specify values for only one field (<code>Pricebook2Id</code> or <code>PricebookId</code>)—not both fields. For this reason, both fields are declared nillable.</p>
PricebookId	<p>Type reference</p>

Field	Field Type
	<p>Properties Create, Defaulted on create, Filter, Nillable, Update</p> <p>Description Unavailable as of version 3.0. As of version 8.0, the Pricebook object is no longer available. Use the <code>Pricebook2Id</code> field instead, specifying the ID of the Pricebook2 record.</p>
Probability	<p>Type percent</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description Percentage of estimated confidence in closing the opportunity. It is implied, but not directly controlled, by the <code>StageName</code> field. You can override this field to a different value than what is implied by the <code>StageName</code>.</p> <p> Note: If you're changing the <code>Probability</code> field through the API using a partner WSDL call, or an Apex <code>before</code> trigger, and the value may have several decimal places, we recommend rounding the value to a whole number. For example, the following Apex in a <code>before</code> trigger uses the <code>round</code> method to change the field value: <code>o.probability = o.probability.round();</code></p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to this object.</p>
StageName	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Current stage of this record. The <code>StageName</code> field controls several other fields on an opportunity. Each of the fields can be directly set or implied by changing the <code>StageName</code> field. In addition, the <code>StageName</code> field is a picklist, so it has additional members in the returned <code>describeSObjectResult</code> to indicate how it affects the other fields. To obtain the stage name values in the picklist, query the <code>OpportunityStage</code> object. If the <code>StageName</code> is updated, then the <code>ForecastCategoryName</code>, <code>IsClosed</code>, <code>IsWon</code>, and <code>Probability</code> are automatically updated based on the stage-category mapping.</p>
SyncedQuoteID	<p>Type reference</p>

Field	Field Type
	<p>Properties Create, Filter, Nillable, Update</p> <p>Description Read only in an Apex trigger. The ID of the Quote that syncs with the opportunity. Setting this field lets you start and stop syncing between the opportunity and a quote. The ID has to be for a quote that is a child of the opportunity.</p>
Territory2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the territory that is assigned to the opportunity. Available only if Enterprise Territory Management has been enabled for your organization.</p>
TotalOpportunityQuantity	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Number of items included in this opportunity. Used in quantity-based forecasting.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Type of opportunity. For example, Existing Business or New Business. Label is Opportunity Type.</p>



Note: If you are importing Opportunity data and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself..

Usage

Use the Opportunity object to manage information about a sale or pending deal. You can also sync this object with a child Quote. To update an Opportunity, your client application needs “Edit” permission on opportunities. You can create, update, delete, and query Attachment records associated with an opportunity via the API. To split credit for an opportunity among multiple opportunity team members, use the OpportunitySplit object.

Client applications can also create or update opportunity objects by converting a Lead with `convertLead()`.



Note: On opportunities and opportunity products, the workflow rules, validation rules, and Apex triggers fire when an update to a child opportunity product or schedule causes an update to the parent record. This means your custom application logic is enforced when there are updates to the parent record, ensuring higher data quality and compliance with your organization's business policies.

Sample Code—Java

This code starts the sync between an object and a child quote.

```
public void startQuoteSync() {
    Opportunity opp = new Opportunity();
    opp.setId(new ID("006D000000CpOSy"));
    opp.setSyncedQuoteId(new ID("0Q0D000000002OZ"));
    // Invoke the update call and save the results
    try {
        SaveResult[] saveResults = binding.update(new SObject[] {opp});
        // check results and do more processing after the update call ...
    }
    catch (Exception ex) {
        System.out.println("An unexpected error has occurred." + ex.getMessage());
        return;
    }
}
```

This code stops the sync between an object and a child quote.

```
public void stopQuoteSync() {
    Opportunity opp = new Opportunity();
    opp.setId(new ID("006D000000CpOSy"));
    opp.setFieldsToNull(new String[] {"SyncedQuoteId"});
    // Invoke the update call and save the results
    try {
        SaveResult[] saveResults = binding.update(new SObject[] {opp});
        // check results and do more processing after the update call ...
    }
    catch (Exception ex) {
        System.out.println("An unexpected error has occurred." + ex.getMessage());
        return;
    }
}
```

SEE ALSO:

[OpportunityCompetitor](#)

[OpportunityHistory](#)

[OpportunityLineItem](#)

[OpportunityLineItemSchedule](#)

[OpportunityFieldHistory](#)

[Quote](#)

[QuoteLineItem](#)

[PartnerNetworkConnection](#)

OpportunityCompetitor

Represents a competitor on an Opportunity.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
CompetitorName	<p>Type combobox</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Name of the competitor.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the associated Opportunity.</p>
Strengths	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Description of the competitor's strengths. Limit: 1,000 characters.</p>

Field	Details
Weaknesses	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the competitor's weaknesses. Limit: 1,000 characters.</p>

Usage

Use this object to manage competitors on an Opportunity, associating multiple competitors on a opportunity and specifying the strengths and weaknesses of each competitor.

SEE ALSO:

[Opportunity](#)

OpportunityContactRole

Represents the role that a Contact plays on an Opportunity.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of an associated Contact. The API applies user access rights to the associated Opportunity for this object, but not to the associated Contact. The API may return rows from a query on this object that include this field's values for contacts to which the user does not have sufficient access rights. It may also return values for this field for contacts that have been deleted. In either case, the client must perform a query on the contact table for this field's value to determine whether the Contact is accessible to the user and has not been deleted.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the associated Contact plays the primary role on the Opportunity (<code>true</code>) or not (<code>false</code>). Each Opportunity has only one primary contact. Label is Primary.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort,</p> <p>Description Required. ID of an associated Opportunity. This field is non-nullable, and it cannot be updated. You must provide a value for this field when creating new records. You can't change it after it has been created.</p>
Role	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Nillable, Update</p> <p>Description Name of the role played by the associated Contact on the Opportunity, such as Business User or Decision Maker.</p>

Usage

Records of this object type appear in the user interface on the Opportunity detail page. Like most other objects, records of this object type have their own unique ID that you use when updating or deleting records.

Although allowed, we do not recommend that you create multiple relationships between the same Opportunity and a Contact.

SEE ALSO:

[Object Basics](#)

OpportunityFeed

Represents a single feed item in the feed displayed on the detail page for an opportunity record. This object is available in API version 18.0 and later.

An opportunity feed shows recent changes to an opportunity record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to opportunities.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Opportunity object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of OpportunityFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your</p>

Field	Details
	<p>comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>

Field	Details
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.

Field	Details
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the opportunity record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nullable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nullable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used.

Field	Details
	<ul style="list-style-type: none"> • <code>CanvasPost</code>—a post made by a canvas app posted on a feed. • <code>CollaborationGroupCreated</code>—generated when a user creates a public group. • <code>CollaborationGroupUnarchived</code>—Not used. • <code>ContentPost</code>—a post with an attached file. • <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher. • <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold. • <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed. • <code>LinkPost</code>—a post with an attached URL. • <code>PollPost</code>—a poll posted on a feed. • <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile. • <code>QuestionPost</code>—generated when a user posts a question. • <code>ReplyPost</code>—generated when Chatter Answers posts a reply. • <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p>

Field	Details
	<ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for an opportunity record.

SEE ALSO:

[Opportunity](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

OpportunityFieldHistory

Represents the history of changes to the values in the fields of an opportunity. This object is available in versions 13.0 and later.

Supported Calls

`describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Opportunity. Label is Opportunity ID.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Usage

Use this object to identify changes to any fields on an Opportunity. The `OpportunityHistory` object represents the history of a change to the `Amount`, `Probability`, `Stage`, or `Close Date` fields of an Opportunity.

This object respects field level security on the parent object.

SEE ALSO:

[Opportunity](#)

OpportunityHistory

Represents the stage history of an Opportunity.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Amount	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Estimated total sale amount.</p>
CloseDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Date when the opportunity is expected to close.</p>
ExpectedRevenue	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Calculated revenue based on the <code>Amount</code> and <code>Probability</code> fields.</p>
ForecastCategory	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Category that determines the column in which an opportunity is totaled in a forecast. Label is To ForecastCategory.</p>
IsDeleted	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the associated Opportunity.</p>
Probability	<p>Type percent</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Percentage of estimated confidence in closing the opportunity.</p>
StageName	<p>Type picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description Name of the current stage of the opportunity (for example, Prospect or Proposal).</p>

Usage

This object represents the history of a change to the `Amount`, `Probability`, `Stage`, or `Close Date` fields of an Opportunity. The `OpportunityFieldHistory` object represents the history of a change to any of the fields of an Opportunity. To obtain information about how a particular opportunity is progressing, query the `OpportunityHistory` records associated with a given Opportunity. Please note that if an opportunity's `Amount`, `Probability`, `Stage`, or `Close Date` fields have not changed, nothing will be returned in the `OpportunityHistory` objects. In this case, query the `OpportunityFieldHistory` records associated with a given Opportunity to get more information about changes to the opportunity.

This object is read-only. The system generates a new record whenever a user or client application changes the value of any of the above fields; the then-current values of all of these major fields are saved in the newly-generated object.

This object respects field-level security on the parent object.

 **Note:** The record is automatically deleted if its parent Opportunity is deleted.

SEE ALSO:

[Opportunity](#)

OpportunityLineItem

Represents an opportunity line item, which is a member of the list of Product2 products associated with an Opportunity.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

The user must have the "Edit" permissions on Opportunity records in order to create or update opportunity line items on an opportunity.

Fields

Field	Details
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
CurrencyIsoCode	<p>Type picklist</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Restricted picklist</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. If the organization has multicurrency enabled, and a Pricebook2 is specified on the parent opportunity (that is, the <code>Pricebook2Id</code> field is not blank on the opportunity referenced by this object's <code>OpportunityId</code>), then the value of this field must match the currency of the <code>CurrencyIsoCode</code> field on the <code>PricebookEntry</code> records that are associated with this object.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Text description of the opportunity line item. Limit: 80 characters.</p>
Discount	<p>Type percent</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Discount for the product as a percentage. When updating these records:</p> <ul style="list-style-type: none"> If you specify <code>Discount</code> without specifying <code>TotalPrice</code>, the <code>TotalPrice</code> is adjusted to accommodate the new <code>Discount</code> value, and the <code>UnitPrice</code> is held constant. If you specify both <code>Discount</code> and <code>Quantity</code>, you must also specify either <code>TotalPrice</code> or <code>UnitPrice</code> so the system knows which one to automatically adjust.
HasQuantitySchedule	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group</p> <p>Description Read-only. Indicates whether a quantity schedule has been created for this object (<code>true</code>) or not (<code>false</code>).</p>
HasRevenueSchedule	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Sort</p> <p>Description Indicates whether a revenue schedule has been created for this object (<code>true</code>) or not (<code>false</code>). If this object has a revenue schedule, the <code>Quantity</code> and <code>TotalPrice</code> fields can't be updated. In addition, the <code>Quantity</code> field can't be updated if this object has a quantity schedule. Update requests aren't rejected but the updated values are ignored.</p>
HasSchedule	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If either <code>HasQuantitySchedule</code> or <code>HasRevenueSchedule</code> is <code>true</code>, this field is also <code>true</code>.</p>
ListPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Corresponds to the <code>UnitPrice</code> on the PricebookEntry that is associated with this line item, which can be in the standard pricebook or a custom pricebook. A client application can use this information to show whether the unit price (or sales price) of the line item differs from the pricebook entry list price.</p>
Name	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The opportunity line item name (known as "Opportunity Product" in the user interface). This read-only field is available in API version 30.0 and later.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the associated Opportunity.</p>

Field	Details
PricebookEntryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the associated PricebookEntry. Exists only for those organizations that have Products enabled as a feature. You can specify values for either this field or <code>ProductId</code>, but not both. For this reason, both fields are declared nillable.</p>
ProductId	<p>Type reference</p> <p>Properties Create, Filter, Nillable</p> <p>Description ID of the related Product record. This field is unavailable as of version 3.0 and is only provided for backward compatibility. The Product object is unavailable beginning with version 8.0. Use the <code>PricebookEntryId</code> field instead, specifying the ID of the PricebookEntry record.</p>
Product2Id	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the related Product2 record. This is a read-only field available in API version 30.0 and later. Use the <code>PricebookEntryId</code> field instead, specifying the ID of the PricebookEntry record.</p>
ProductCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This read-only field is available in API version 30.0 and later. It references the value in the <code>ProductCode</code> field of the related <code>Product2</code> record.</p>
Quantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p>

Field	Details
	<p>Description</p> <p>Read-only if this record has a quantity schedule, a revenue schedule, or both a quantity and a revenue schedule.</p> <p>When updating these records:</p> <ul style="list-style-type: none"> • If you specify <code>Quantity</code> without specifying the <code>UnitPrice</code>, the <code>UnitPrice</code> value will be adjusted to accommodate the new <code>Quantity</code> value, and the <code>TotalPrice</code> will be held constant. • If you specify both <code>Discount</code> and <code>Quantity</code>, you must also specify either <code>TotalPrice</code> or <code>UnitPrice</code> so the system can determine which one to automatically adjust.
RecalculateTotalPrice	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Changes behavior of OpportunityLineItem calculations when a line item has child schedule rows for the <code>Quantity</code> value. When enabled, if the rollup quantity changes, then the quantity rollup value is multiplied against the sales price to change the total price. Product2 flag must be set to true.</p>
ServiceDate	<p>Type</p> <p>date</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>Date when the product revenue will be recognized and the product quantity will be shipped. Whether this value is used by customizable forecasting depends upon the Forecast Date setting for the organization:</p> <ul style="list-style-type: none"> • Opportunity Close Date—<code>ServiceDate</code> is ignored. • Product Date—<code>ServiceDate</code> is used if not null. • Schedule Date—<code>ServiceDate</code> is used if not null and there are no revenue schedules present for this line item, that is, there are no <code>OpportunityLineItemSchedule</code> records with a field <code>Type</code> value of Revenue that are children of this record.
SortOrder	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Number indicating the sort order selected by the user. Client applications can use this to match the sort order in Salesforce.</p>

Field	Details
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description Difference between standard and discounted pricing. Converted currency amounts when the opportunity's currency is different from the user's currency.</p>
TotalPrice	<p>Type currency</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description This field is available only for backward compatibility. It represents the total price of the OpportunityLineItem. If you do not specify <code>UnitPrice</code>, this field is required. If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>UnitPrice</code> is required. When updating these records, you can change either this value or the <code>UnitPrice</code>, but not both at the same time. This field is nillable, but you can't set both <code>TotalPrice</code> and <code>UnitPrice</code> to null in the same update request. To insert the <code>TotalPrice</code> via the API (given only a unit price and the quantity), calculate this field as the unit price multiplied by the quantity. This field is read-only if the opportunity line item has a revenue schedule. If the opportunity line item does not have a schedule or only has quantity schedule, this field can be updated.</p>
UnitPrice	<p>Type currency</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description The unit price for the opportunity line item. In the Salesforce user interface, this field's value is calculated by dividing the total price of the opportunity line item by the quantity listed for that line item. Label is Sales Price. This field or <code>TotalPrice</code> is required. You can't specify both. If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>TotalPrice</code> is required.</p>

Usage

An Opportunity can have associated OpportunityLineItem records only if the Opportunity has a Pricebook2. An OpportunityLineItem must correspond to a Product2 that is listed in the opportunity's Pricebook2. For information about inserting OpportunityLineItem for an opportunity that does not have an associated Pricebook2 or any existing line items, see [Effects on Opportunities](#).

This object is defined only for those organizations that have products enabled as a feature. If the organization does not have the products feature, this object does not appear in the `describeGlobal()` call, and you can't use `describeSObjects()` or query the `OpportunityLineItem` object.

For a visual diagram of the relationships between `OpportunityLineItem` and other objects, see [Product and Schedule Objects](#).



Note: If multicurrency option has been enabled, the `CurrencyIsoCode` field is present. It can't be modified, and is always set to the value of the `CurrencyIsoCode` of the parent Opportunity.

Effects on Opportunities

Opportunities that have associated `OpportunityLineItem` records are affected in the following ways:

- Creating an `OpportunityLineItem` increments the `Opportunity Amount` value by the `TotalPrice` of the `OpportunityLineItem`. Additionally, inserting an `OpportunityLineItem` increments the `ExpectedRevenue` on the opportunity by the `TotalPrice` times the opportunity `Probability`.
- The `Opportunity Amount` becomes a read-only field when the opportunity has line items. The API ignores any attempt to update this field on an opportunity with line items. Update requests are not rejected, but the updated value is ignored.
- You can't update the `PricebookId` field or the `CurrencyIsoCode` field on the opportunity if line items exist. The API rejects any attempt to update these fields on an opportunity with line items.
- When you create or update an `OpportunityLineItem`, the API verifies that the line item corresponds to a `PricebookEntry` in the `Pricebook2` that is associated with the opportunity. If the opportunity does not have an associated `Pricebook2`, the API automatically sets the pricebook on the opportunity if the line item corresponds to a `PricebookEntry` in an active `Pricebook2`, and if the `PricebookEntry` has a `CurrencyIsoCode` field that matches the `CurrencyIsoCode` field of the opportunity. If the `Pricebook2` is not active or the `CurrencyIsoCode` fields do not match, an error is returned.
- The `Opportunity HasOpportunityLineItem` field is set to `true` when an `OpportunityLineItem` is inserted for that Opportunity.

SEE ALSO:

[OpportunityLineItemSchedule](#)

OpportunityLineItemSchedule

Represents information about the quantity, revenue distribution, and delivery dates for a particular `OpportunityLineItem`.

In API version 38.0 and later, when an `OpportunityLineItem` record is created for a product with a previously established schedule, an `OpportunityLineItemSchedule` record is also created.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Restricted picklist, Update</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. This field is available in version 10.0 and later.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of the opportunity line item schedule. Limit: 80 characters. Label is Comments.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityLineItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the associated OpportunityLineItem.</p>
Quantity	<p>Type double</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Required. The total number of units to be scheduled in a quantity schedule.</p>

Field	Details
Revenue	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The revenue that should be recognized, or the quantity that should be shipped, or both - depending upon the value of Type.</p>
ScheduleDate	<p>Type date</p> <p>Properties Create, Filter, Update</p> <p>Description Required. The date the associated OpportunityLineItem is to be scheduled for an event: delivery, shipping, or any other date you wish to track.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description The type of the schedule. Required when inserting an OpportunityLineItemSchedule. Valid values include Quantity, Revenue, or Both.</p>

Allowed Type Field Values

The allowed **Type** values for an OpportunityLineItemSchedule depend on the product-level schedule preferences and whether the line item has any existing schedules. The following criteria must be met:

- The Product2 on which the OpportunityLineItem is based must have the appropriate **CanUseRevenueSchedule** or **CanUseQuantitySchedule** fields (or both) set to `true`.
- When you create a schedule for a line item that does not have any existing schedules, you can specify any valid value.
- If you create a schedule for a line item that already has existing schedules, the new schedule must be consistent with the existing schedules. The following matrix outlines the allowable values:

Value of HasRevenueSchedule on line item	Value of HasQuantitySchedule on line item	Allowable Type Values
false	false	Revenue, Quantity, both
false	true	Quantity
true	false	Revenue

Value of HasRevenueSchedule on line item	Value of HasQuantitySchedule on line item	Allowable Type Values
true	true	both

Allowed Quantity and Revenue Field Values

The allowable Quantity and Revenue field values depend on the value of the Type field:

Type Value	Allowable Quantity Value	Allowable Revenue Value
Revenue	Null	Non-null
Quantity	Non-null	Null
both	Non-null	Non-null

The Quantity and Revenue fields have the following restrictions when this object is updated:

- For a schedule of Type Quantity, you can't update a null Revenue value to non-null. Likewise for a schedule of Type Revenue, you can't update a null Quantity value to non-null.
- You can't null out the Quantity field for a schedule of Type Quantity. Likewise you can't null out the Revenue field for a schedule of Type Revenue.
- You can't null out either the Revenue or Quantity fields for a schedule of type Both.

Usage

OpportunityLineItemSchedule supports two types of schedules:

- Quantity schedules
- Revenue schedules

The user must have edit access rights on the Opportunity in order to create or update line item schedules on that opportunity.

Products and Schedules Must Be Enabled

The OpportunityLineItemSchedule object is defined only for those organizations that have the products and schedules features enabled. If the organization does not have the products and schedules features, the OpportunityLineItemSchedule object is not returned in a describe, and you can't describe or query OpportunityLineItemSchedule records.

Effects on Opportunities and Opportunity Line Items

OpportunityLineItemSchedule records affect opportunities and opportunity line items in the following ways:

- Inserting an OpportunityLineItemSchedule of Type "Revenue" or "Quantity" increments the TotalPrice field on the OpportunityLineItem by the OpportunityLineItemSchedule Revenue amount. Inserting an OpportunityLineItemSchedule of Type Quantity or Both increments the Quantity field on the OpportunityLineItem by the OpportunityLineItemSchedule Quantity amount.
- Creating an OpportunityLineItemSchedule record affects the original opportunity:

1. The Opportunity `Amount` is incremented by the `OpportunityLineItemSchedule` revenue amount
 2. The Opportunity `ExpectedRevenue` is incremented by the line item schedule amount multiplied by the Opportunity `Probability`
- Deleting an `OpportunityLineItemSchedule` has a similar effect on the related `OpportunityLineItem` and `Opportunity`. Deleting an `OpportunityLineItemSchedule` decrements the `OpportunityLineItem TotalPrice` by the deleted `OpportunityLineItemSchedule Quantity` or `Revenue` amount. The Opportunity `Amount` is also decremented by the `OpportunityLineItemSchedule Quantity` or `Revenue` amount, and the Opportunity `ExpectedRevenue` is reduced by `OpportunityLineItemSchedule Quantity` or `Revenue` amount multiplied by the Opportunity `Probability`.

Deleting an Opportunity Line Item Schedule

Deleting the last remaining schedule will set the corresponding `HasQuantitySchedule` or `HasRevenueSchedule` flags (or both) to `false` on the parent line item.

SEE ALSO:

[OpportunityLineItem](#)

[Product2](#)

[delete\(\)](#)

OpportunityOverride

Represents a forecast override of an Opportunity. This read-only object is specific to customizable forecasting. It has a parent-child relationship with `LineitemOverride`.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Requires the “View All Data” permission.

Fields

Field	Details
<code>AmountInherited</code>	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden amount rolls up through the forecast hierarchy (<code>true</code>), or was overridden by the owner of the <code>OpportunityOverride</code> (<code>false</code>).</p>

Field	Details
ForecastCategoryInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden forecast category rolls up through the forecast hierarchy (<code>true</code>) or was overridden by the owner of the OpportunityOverride (<code>false</code>).</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the associated Opportunity.</p>
OutOfDate	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> if a subordinate user's OpportunityOverride, or the opportunity itself, has been updated since this override was last updated, such that an overridden value on this override may be obsolete. For example, a subordinate user has more recently overridden the same period, category, or amount field.</p>
OverrideAmount	<p>Type currency</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The total monetary amount of the opportunity, possibly overridden.</p>

Field	Details
OverrideComment	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The comment entered on the opportunity forecast edit page. Limit: 255 characters.</p>
OverrideForecastCategory	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The forecast category of the opportunity, possibly overridden.</p>
OverridePeriodId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the associated fiscal Period, possibly overridden. If you are using custom fiscal years and the period falls after your custom fiscal year has ended, then the override is not returned.</p>
OverrideQuantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The quantity of the opportunity, possibly overridden.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the OpportunityOverride owner.</p>
PeriodInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Indicates whether the overridden period rolls up through the forecast hierarchy (<code>true</code>) or was overridden by the OpportunityOverride owner (<code>false</code>).</p>
QuantityInherited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the overridden quantity rolls up through the forecast hierarchy (<code>true</code>) or was overridden by the OpportunityOverride owner (<code>false</code>).</p>

 **Important:** You can't use the `<apex:relatedList>` component to add related lists such as OpportunityForecast to Visualforce pages because the `RelationshipName` field of OpportunityOverride is not exposed in the API. Use the `<apex:detail>` component instead.

SEE ALSO:

[LineitemOverride](#)

OpportunityOwnerSharingRule

Represents a rule for sharing an opportunity with users other than the owner.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. Opportunities owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field	Details
	<p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group. The target user or group is being given access.</p>

Usage

Use this object to manage the sharing rules for opportunities. General sharing and Territory-related sharing use this object.

SEE ALSO:

[Case](#)

[Metadata API Developer Guide: SharingRules](#)

OpportunityPartner

This read-only object represents a partner relationship between an Account and an Opportunity. This object is automatically created when a Partner object is created for a partner relationship between an account and an opportunity.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field	Details
AccountToId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the partner Account in the partner relationship.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the account is the opportunity's primary partner (<code>true</code>) or not (<code>false</code>). Label is Primary.</p>
OpportunityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Opportunity that is in the partner relationship.</p>
Role	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The UserRole that the Account has on the Opportunity. For example, <code>Reseller</code> or <code>Manufacturer</code>.</p>

Creating an Account-Opportunity Partner Relationship

When you create a partner relationship between an account and an opportunity (when you create a Partner object and specify the `OpportunityId` field), the API automatically creates an `OpportunityPartner` with the corresponding values:

- The value of the `Partner` field `AccountId` maps to the value of the `OpportunityPartner` field `AccountToId`.
- The values of the `OpportunityId`, `Role`, and `IsPrimary` fields in both objects are the same.
- If you set the `IsPrimary` value to 1 (`true`) upon insert of a new `OpportunityPartner`, any other existing primary partners for that opportunity will automatically have the `IsPrimary` value set to 0 (`false`).

This mapping allows the API to manage the objects and their relationship efficiently.

SEE ALSO:

[Partner](#)

[AccountPartner](#)

OpportunityShare

Represents a sharing entry on an Opportunity.

Supported Calls

`describeSObjects(), create(), delete(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Level of access that the user or group has to the opportunity. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All—This value is not valid when creating, updating, or deleting records. <p>This field must be set to an access level that's higher than the org's default access level for opportunities.</p>
OpportunityId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the opportunity associated with this sharing entry. This field can't be updated.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Valid values include:</p> <ul style="list-style-type: none"> • <code>Owner</code>—User is the owner of the opportunity. • <code>Manual</code>—User or group has access because a user with “All” access manually shared the opportunity with the user or group. • <code>Rule</code>—User or group has access via an opportunity sharing rule. • <code>ImplicitChild</code>—User or group has access to the opportunity on the account associated with this opportunity. • <code>Sales Team</code>—User has access to the opportunity because the user is on the opportunity sales team for the opportunity. The <code>OpportunityTeamMember</code> object sets the access level. See <code>OpportunityTeamMember</code> for more information.
UserOrGroupId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user or group that has been given access to the opportunity. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit opportunities owned by other users.

If you attempt to create a record that matches an existing record, any modified fields are updated, the system returns the existing record.

If an opportunity is shared in multiple ways with a user, you don't always see multiple sharing records. If a user has access to an opportunity for one or more of the following RowCause values, the records in the OpportunityShare object are compressed into one record with the highest level of access.

- `Manual`

- Owner

SEE ALSO:

[Object Basics](#)

OpportunitySplit

OpportunitySplit credits one or more opportunity team members with a portion of the opportunity amount. This object is available in API version 16.0 and later for pilot customers, and version 28.0 and later for others.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
OpportunityId	<p>Type reference</p> <p>Properties Create, FilterGroup, Sort</p> <p>Description ID of the opportunity for which the split is being created. Label is Opportunity ID.</p>
Split	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description Read-only. Automatically-generated number identifying the split within the opportunity.</p>
SplitAmount	<p>Type currency</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Monetary amount of the split. Label is Split Amount.</p>

Field	Details
SplitNote	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Enter any notes or comments about the split. Character limit is 255. Label is Split Note.</p>
SplitOwnerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The opportunity owner. Label is User ID.</p>
SplitPercentage	<p>Type percent</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description Split percentage that this team member will receive. If the split type is validated to a 100% total, this number can range from 0 to 100. If the total isn't validated, this number can range from 0 to 1,000. Label is Split (%).</p>
SplitTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Auto-generated, numeric ID for the split type defined by the OpportunitySplitType object. This field is available in API version 28 and later. If this field is blank, the system automatically specifies the default split type for the opportunity amount, which is validated to 100%.</p>

Usage

Use the OpportunitySplit object to manage splits for an opportunity.

If you change the opportunity owner using the API, the old owner remains on the opportunity team with either Read-only access, or the level of access specified in your organization-wide defaults.

OpportunitySplitType

OpportunitySplitType provides unique labels and behavior for each split type. This object is available in API version 28.0 and later.

There are two default split types: revenue splits, which must total 100%, and overlay splits, which can total any percentage.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Describes the purpose of the split type, providing context to future developers.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Enables or disables the split type.</p>

Field Name	Details
IsTotalValidated	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the split must total 100%. If <code>false</code>, the split can total any percentage.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates language of split labels in the user interface.</p>
ManageableState	<p>Type ManageableState enumerated list</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates the manageable state of the specified component that is contained in a package:</p> <ul style="list-style-type: none"> • <code>beta</code> • <code>deleted</code> • <code>deprecated</code> • <code>installed</code> • <code>released</code> • <code>unmanaged</code> <p>For more information about states of manageability for components in Force.com AppExchange packages, see “Planning the Release of Managed Packages” in the Salesforce online help.</p> <p>This field is available in API version 38.0 and later.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The user-interface label for the split type.</p>
NamespacePrefix	<p>Type string</p>

Field Name	Details	
Properties	Filter, Group, Nillable, Sort	
Description	The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.	
The namespace prefix can have one of the following values:	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. 	
This field can't be accessed unless the logged-in user has the "Customize Application" permission.		
SplitEntity	Type	picklist
Properties	Create, Filter, Group, Restricted picklist, Sort	
Description	The containing record type, such as an opportunity. Available in API version 30 and later.	
SplitField	Type	picklist
Properties	Create, Filter, Group, Restricted picklist, Sort	
Description	Indicates which currency field of the opportunity object is split. Available in API version 30 and later.	
SplitDataStatus	Type	picklist
Properties	Filter, Group, Nillable, Restricted picklist, Sort, Update	
Description	Indicates the status of the split type. Available in API version 30 and later.	

OpportunityStage

Represents the stage of an Opportunity in the sales pipeline, such as New Lead, Negotiating, Pending, Closed, and so on.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
DefaultProbability	<p>Type percent</p> <p>Properties Filter, Nillable, Sort,</p> <p>Description The default percentage estimate of the confidence in closing a specific opportunity for this opportunity stage value. Label is Probability (%).</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Description of this opportunity stage value. Limit: 255 characters.</p>
ForecastCategory	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The default forecast category for this opportunity stage value. The forecast category automatically determines how opportunities are tracked and totaled in a forecast.</p>

Field	Details
ForecastCategoryName	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Available in API version 12.0 and later. The default forecast category value for this opportunity stage value.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this opportunity stage value is active (<code>true</code>) or not (<code>false</code>). Inactive opportunity stage values are not available in the picklist and are retained for historical purposes only.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this opportunity stage value represents a closed opportunity (<code>true</code>) or not (<code>false</code>). Multiple opportunity stage values can represent a closed opportunity. Label is Closed.</p>
IsWon	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this opportunity stage value represents a won opportunity (<code>true</code>) or not (<code>false</code>). Multiple opportunity stage values can represent a won opportunity. Label is Won.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this opportunity stage value. This display value is the internal label that does not get translated. Limit: 255 characters.</p>

Field	Details
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the opportunity stage picklist. These numbers are not guaranteed to be sequential, as some previous opportunity stage values might have been deleted.</p>

Usage

This object represents a value in the opportunity stage picklist, which provides additional information about the stage of a Opportunity, such as its probability or forecast category. Query this object to retrieve the set of values in the opportunity stage picklist, and then use that information while processing Opportunity records to determine more information about a given opportunity. For example, the application could test whether a given opportunity is won or not based on its `StageName` value and the value of the `IsWon` property in the associated OpportunityStage object.

This object is read-only via the API.

SEE ALSO:

[Object Basics](#)

OpportunityTag

Associates a word or short phrase with an Opportunity.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>

Field Name	Details
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

OpportunityTag stores the relationship between its parent TagDefinition and the Opportunity being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

OpportunityTeamMember

Represents a User on the opportunity team of an Opportunity.

See also UserTeamMember, which represents a User who is on the default opportunity team of another user.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the org.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p> <p> Note: An OpportunityTeamMember record that is deleted is not moved to the Recycle Bin and cannot be undeleted, unless the record was cascade-deleted when deleting a related Opportunity. For directly deleted OpportunityTeamMember records, don't use the <code>isDeleted</code> field to detect deleted records in SOQL queries or <code>queryAll()</code> calls. Instead, use <code>getDeleted()</code>.</p>
Name	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The team member name. This read-only field is available in API version 30.0 and later.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist</p> <p>Description Opportunity access level for this team member. Valid values:</p> <ul style="list-style-type: none"> • Read

Field	Details
	<ul style="list-style-type: none">• Edit <p>This field is supported in triggers, but not in workflows or validation rules. It is editable in API version 36.0 and later.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the Opportunity associated with this opportunity team. This field can't be updated.</p>
PhotoURL	<p>Type URL</p> <p>Properties Filter, Nullable, Sort, Group</p> <p>Description Read only. Retrieves the users Chatter photo URL. This field is available in API version 32.0 and later.</p>
TeamMemberRole	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Role that the team member has on the opportunity. The org's admin sets the valid values in the Opportunity Team Roles picklist. Label is Team Role.</p>
Title	<p>Type string</p> <p>Properties Filter, Nullable, Sort, Group</p> <p>Description Read only. Retrieves the user's title. This field is available in API version 36.0 and later.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the User who is a member of the opportunity team. This field can't be updated.</p>

Usage

If you create a record for this object and it matches an existing record, the system updates any modified fields and returns the existing record.

In the user interface, users can set up an opportunity team for the opportunities they own. The opportunity team includes other users that are working on the opportunity with them. This object is available only in organizations that have enabled team selling.

-  **Note:** The behavior for changing ownership of opportunities is different using the user interface when the previous owner is on an opportunity team. For example, when you change the owner of an opportunity using the API, the previous owner's access becomes Read Only or the access specified in your organization-wide default for opportunities, whichever is greater. However, performing this same action in the user interface allows you to select the access level for the previous owner when the previous owner is on an opportunity team.

SEE ALSO:

[UserTeamMember](#)

Order

Represents an order associated with a contract or an account.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Group, Sort, Update</p> <p>Description Required. ID of the Account associated with this order. Can only be updated when the order's <code>StatusCode</code> value is <i>Draft</i>.</p>
ActivatedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the User who activated this order.</p>

Field Name	Details
ActivatedDate	Type dateTime
	Properties Filter, Nillable, Sort, Update
	Description Date and time when the order was activated.
BillingCity	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description City for the billing address for this order. Maximum size is 40 characters.
BillingCountry	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Country for the billing address for this order. Maximum size is 80 characters.
BillingCountryCode (beta)	Type picklist
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description ISO country code for the billing address for this order.
BillingLatitude	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
BillingLongitude	Type double
	Properties Create, Filter, Nillable, Sort, Update

Field Name	Details
	<p>Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
BillingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal code for the billing address for this order. Maximum size is 20 characters.</p>
BillingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the billing address for this order. Maximum size is 80 characters.</p>
BillingStateCode (beta)	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO state code for the order's billing address.</p>
BillingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street address for the billing address.</p>
BillToContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the contact that the order will be billed to.</p>
CompanyAuthorizedById	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description ID of the user who authorized the account associated with the order.</p>
CompanyAuthorizedDate	<p>Type date</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description Date on which the order was authorized by your organization.</p>
ContractId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description ID of the contract associated with this order. Can only be updated when the order's <code>statusCode</code> value is <i>Draft</i>.</p>
CustomerAuthorizedById	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description ID of the contact who authorized the order.</p>
CustomerAuthorizedDate	<p>Type date</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description Date on which the contact authorized the order.</p>
Description	<p>Type textarea</p>
	<p>Properties Create, Nillable, Update</p>
	<p>Description Description of the order.</p>

Field Name	Details
EffectiveDate	<p>Type date</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Date at which the order becomes effective. Label is Order Start Date.</p>
End Date	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date at which the order ends. Label is Order End Date.</p>
IsReductionOrder	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. Determines whether an order is a reduction order. Label is Reduction Order.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
Name	<p>Type string</p>

Field Name	Details
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description Name for this order.
OpportunityId	Type reference
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description ID for the opportunity that's associated with this order.
OrderNumber	Type string
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description Order number assigned to this order (not the unique, system-generated ID assigned during creation). Maximum size is 30 characters.
OrderReferenceNumber	Type string
	Properties Create, Filter, Group, Nullable, Sort, Update
	Description Order reference number assigned to this order. Maximum size is 80 characters.
OriginalOrderId	Type reference
	Properties Create, Defaulted on create, Filter, Group, Nullable, Sort
	Description Optional. ID of the original order that a reduction order is reducing, if the reduction order is reducing a single order. Label is Original Order .
	Editable only if <code>isReductionOrder</code> is <code>true</code> . If the reduction order is reducing more than one order, leave blank.
OwnerId	Type reference
	Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	<p>Description Required. ID of the User or queue that owns this order.</p>
PoDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Date of the purchase order.</p>
PoNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Number identifying the purchase order. Maximum is 80.</p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description Required. ID of the price book associated with this order.</p>
QuoteId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the quote that's associated with this order. If you set QuoteId to null, QuoteLineItemId on all of the order's child order products is set to null.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the record type assigned to this order.</p>
ShippingCity	<p>Type string</p>

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description City of the shipping address. Maximum size is 40 characters.
ShippingCountry	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Country of the shipping address. Maximum size is 80 characters.
ShippingCountryCode (beta)	Type picklist
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description ISO country code for the order's shipping address.
ShippingLatitude	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
ShippingLongitude	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
ShippingPostalCode	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	<p>Description Postal code of the shipping address. Maximum size is 20 characters.</p>
ShippingState	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description State of the shipping address. Maximum size is 80 characters.</p>
ShippingStateCode (beta)	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description ISO state code for the order's shipping address.</p>
ShippingStreet	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description Street address of the shipping address. Maximum of 255 characters.</p>
ShipToContactId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description ID of the contact the order will be shipped to.</p>
Status	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
	<p>Description Picklist of values that indicate order status. Each value is within one of two status categories defined in StatusCode. For example, the status picklist might contain <i>Draft</i>, <i>Ready for Review</i>, and <i>Ready for Activation</i> values with a StatusCode of <i>Draft</i>.</p>

Field Name	Details
StatusCode	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status category for the order. An order can be either <i>Draft</i> or <i>Activated</i>. Label is Status Category.</p>
TotalAmount	<p>Type currency</p> <p>Properties Filter, Sort</p> <p>Description The total amount for the order products associated with this order.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The type of order.</p>

Usage

The `Status` field specifies the current state of an order. Status strings represent its current state (*Draft* or *Activated*).

When a client application creates an order, the `Status Code` must be *Draft* and the `Status` must be any value that corresponds to a `Status Code` of *Draft*. The application can subsequently activate an order by updating it and setting the value in its `Status` field to an *Activated* state; however, the `Status` field is the only field you can update when activating the order.

After an order is activated, your client application can change the `Status` back to the *Draft* state—but only if the order doesn't have any child reduction order products. Your client application can delete orders when the `Status` is *Draft* but not when its `Status` is *Activated*.

Client applications can use the API to create, update, delete, and query any Attachment associated with an order.

SEE ALSO:

- [OrderFeed](#)
- [OrderHistory](#)
- [OrderItem](#)

OrderFeed

Represents a single feed item in the feed displayed on an order.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	Type textarea Properties Nillable, Sort Description The content of the OrderFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.
CommentCount	Type int Properties Filter, Group, Sort Description The number of FeedComments associated with this feed item. Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.

Field Name	Details
	<p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Create, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nullable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>

Field Name	Details
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p>
	<p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities.
	<p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the order record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of FeedItem:<ul style="list-style-type: none">• ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.</p>

Field Name	Details
	<ul style="list-style-type: none"> ● <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher. ● <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold. ● <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed. ● <code>LinkPost</code>—a post with an attached URL. ● <code>PollPost</code>—a poll posted on a feed. ● <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile. ● <code>QuestionPost</code>—generated when a user posts a question. ● <code>ReplyPost</code>—generated when Chatter Answers posts a reply. ● <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com. ● <code>TextPost</code>—a direct text entry on a feed. ● <code>TrackedChange</code>—a change or group of changes to a tracked field. ● <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> ● <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. ● <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. ● <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. ● <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. ● <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. ● <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. ● <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. ● <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. ● <code>SocialPost</code>—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Field Name	Details
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none">• AllUsers—The feed item is available to all users who have permission to see the feed item.• InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none">• For record posts, Visibility is set to InternalUsers for all internal users by default.• External users can set Visibility only to AllUsers.• Visibility can be updated on record posts.• The Update property is supported only for feed items posted on records.

Usage

Use this object to track changes for an order record.

SEE ALSO:

[Order](#)

OrderHistory

Represents historical information about changes that have been made to the standard fields of the associated order, or to any custom fields with history tracking enabled.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Name of the order field that was modified, or a special value to indicate some other modification to the order.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description New value of the modified order field. Maximum of 255 characters.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description Previous value of the modified order field. Maximum of 255 characters.</p>
OrderId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the order associated with this record.</p>

Usage

Order history entries are automatically created each time an order is modified.

Two rows are added to this record when foreign key fields change. One row contains the foreign key object names that display in the online application. For example, Jane Doe is recorded as the name of a Contact. The other row contains the actual foreign key ID that is only returned to and visible from the API.

This object respects field-level security on the parent object.

SEE ALSO:

[Order](#)

OrderItem

Represents an order product that your organization sells.

Supported Calls

`create(), delete(), getDeleted(), getUpdated(), query(), retrieve(), update()`

Fields

Field Name	Details
AvailableQuantity	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Amount of an order product that is available to be reduced. Value must be greater than or equal to 0. An order product is reducible only if <code>AvailableQuantity</code> is greater than 0. Value is always 0 if the order product's parent order is a reduction order.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of this object.</p>
EndDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Optional. Last day the order product is available.</p>
ListPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description List price for the order product. Value is inherited from the associated <code>PriceBookEntry</code> upon order product creation.</p>

Field Name	Details
OrderId	Type reference
	Properties Create, Filter, Group, Sort
	Description ID of the order that this order product is a child of.
OrderItemNumber	Type string
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description Automatically-generated number that identifies the order product.
OriginalOrderItemId	Type reference
	Properties Create, Filter, Group, Nillable, Sort
	Description Required if <code>isReductionOrder</code> on the parent order is <code>true</code> . ID of the original order product being reduced.
PricebookEntryId	Type reference
	Properties Create, Filter, Group, Sort
	Description Required. ID of the associated PricebookEntry. This field must be specified when creating OrderItem records. It can't be changed in an update.
Quantity	Type double
	Properties Create, Filter, Sort, Update
	Description Number of units of this order product. If you specify <code>Quantity</code> without specifying <code>UnitPrice</code> , <code>UnitPrice</code> is adjusted to accommodate the new <code>Quantity</code> value while keeping <code>TotalPrice</code> constant.

Field Name	Details
QuoteLineItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the associated quote line item. If this field is specified, the quote line item's QuotId must match the QuotId for the order product's parent order.</p>
ServiceDate	<p>Type date</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Start date for the order product. Label is Start Date.</p>
UnitPrice	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Unit price for the order product. In the Salesforce user interface, this field's value is calculated by dividing the total price of the order product by the quantity listed for that order product.</p>

Usage

An order can have associated order product records only if the order has a price book associated with it. An order product must correspond to a product that is listed in the order's price book.

SEE ALSO:

[OrderItemFeed](#)
[OrderItemHistory](#)
[Order](#)

OrderItemFeed

Represents a single feed item in the feed displayed on the detail page for an order product record.

An order product feed shows changes to an order product for fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to order products in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>

Field Name	Details
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read-only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read-only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then InsertedById is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item Body contains rich text. Set IsRichText to true if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p>

Field Name	Details
	<p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The title of the FeedItem. When the Type is LinkPost, the <code>LinkUrl</code> is the URL and this field is the link name.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of FeedItem:<ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence. • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. </p>

Field Name	Details
	<ul style="list-style-type: none"> • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. • <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Visibility	Type picklist Properties Create , Filter , Group , Nillable , Restricted picklist , Sort , Update Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. Visibility can have the following values: <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. Note the following exceptions for Visibility: <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • <code>Visibility</code> can be updated on record posts.
------------	---

Field Name	Details
	<ul style="list-style-type: none"> The <code>update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for an order product record.

SEE ALSO:

[OrderItem](#)

OrderItemHistory

Represents the history of changes to the values in the fields of an order product.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>

Field Name	Details
OrderItemId	<p>Type reference</p> <p>Properties Group, Sort</p> <p>Description ID of the order product. Label is Order Product ID.</p>

Usage

Use this object to identify changes to an order product.

This object respects field-level security on the parent object.

SEE ALSO:

[OrderItem](#)

OrderOwnerSharingRule

Represents a rule which determines order sharing access for the order's owners.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CreatedBy	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the creator of the order owner sharing rule.</p>

Field	Details
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date when the order owner sharing rule was created.</p>
Description	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Description of the order owner sharing rule. Maximum length is 1,000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Name of the developer of the order owner sharing rule.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the group whose orders are shared.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, idLookup, Sort</p> <p>Description ID of the order owner sharing rule.</p>
LastModifiedById	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the user who last modified the order owner sharing rule.</p>

Field	Details
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date when the order owner sharing rule was last modified.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Namefield, Sort, Update</p> <p>Description Name of the order owner sharing rule. Maximum length is 80 characters.</p>
OrderAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Access level for the order owner sharing rule.</p>
SystemModstamp	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description System modification time for the order owner sharing rule.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group with whom order access is shared.</p>

Usage

Use this object to manage the sharing rules for orders. For example, the following code creates an order owner sharing rule between two public groups, which can also contain portal users.

```
OrderOwnerSharingRule rule = new OrderOwnerSharingRule();
rule.setName("RuleName"); // Set the sharing rule name
```

```
rule.setDeveloperName("RuleDeveloperName"); // Set the sharing rule developer name  
rule.setGroupId("00Gx000000000000"); // Set the group of users to share records from  
rule.setUserOrGroupId("00Gx000000000001"); // Set the group of users to share records to  
rule.setOrderAccessLevel("Edit");  
connection.create(rule);
```

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

Organization

Represents key configuration information for an organization.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`

Executing a SOQL SELECT query returns the value of fields in this object, but no value is visible for some of the fields.

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Address (beta)	Type address Properties Filter, Nillable Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.
AllowsSelfServiceLogin	Type boolean Properties Defaulted on create, Filter, Update Description Indicates whether the organization allows Self-Service login (<code>true</code>) or not (<code>false</code>).
City	Type string

Field	Details
	<p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Name of the city for the organization's address.</p>
ComplianceBccEmail	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Email address for compliance blind carbon copies. Limit: 80 characters.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the country for the organization's address. Limit: 80 characters.</p>
CountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the organization's address.</p>
DailyWebToCaseCount	<p>Type int</p> <p>Properties Filter, Nillable</p> <p>Description The number of web form submissions that have been converted to cases for the day.</p>
DailyWebToCaseLimit	<p>Type int</p> <p>Properties Filter, Nillable</p>

Field	Details
	Description
	The maximum number of web form submissions that can be converted to cases per day.
DailyWebToLeadCount	Type int
	Properties
	Filter, Nillable
	Description
	The number of web form submission that have been converted to leads for the day
DailyWebToLeadLimit	Type int
	Properties
	Filter, Nillable
	Description
	The maximum number of web form submissions that can be converted to leads per day.
DefaultAccountAccess	Type picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	In API version 10.0 and later, represents the default access level for accounts, contracts, and assets. The possible values are:
	<ul style="list-style-type: none"> • None • Read • Edit
	In versions before 10.0, <code>DefaultAccountAndContactAccess</code> represented this value.
DefaultAccountAndContactAccess	Type picklist
	Properties
	Filter, Nillable, Restricted picklist
	Description
	Default access level for accounts, contacts, contracts, and assets. This field is supported for backward compatibility only and is not available in API version 10.0 or later. In version 10.0 and later, use

Field	Details
	either DefaultAccountAccess or DefaultContactAccess.
DefaultCalendarAccess	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description Default access level for calendars. The possible values are listed, followed by the user interface labels in parentheses: <ul style="list-style-type: none">• HideDetails (Hide Details)• HideDetailsInsert (Hide Details and Add Events)• ShowDetails (Show Details)• ShowDetailsInsert (Show Details and Add Events)• AllowEdits (Full Access)
DefaultCampaignAccess	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Default access level for campaigns. The possible values are: <ul style="list-style-type: none">• None• Read• Edit• All
DefaultCaseAccess	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Default access level for cases. The possible values are: <ul style="list-style-type: none">• None• Read• Edit• ReadEditTransfer
DefaultContactAccess	Type picklist

Field	Details
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Default access level for contacts. The possible values are:</p> <ul style="list-style-type: none">• None• Read• Edit• ControlledByParent <p>In versions before 10.0, <code>DefaultAccountAndContactAccess</code> represented this value.</p> <p> Note: When <code>DefaultContactAccess</code> is set to "Controlled by Parent," you can't update the <code>ContactAccessLevel</code> field.</p>
<code>DefaultLeadAccess</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Default access level for leads. The possible values are:</p> <ul style="list-style-type: none">• NoneRead• Edit• ReadEditTransfer
<code>DefaultLocaleSidKey</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Default locale SID key.</p>
<code>DefaultOpportunityAccess</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Default access level for opportunities. The possible values are:</p> <ul style="list-style-type: none">• None• Read

Field	Details
DefaultPricebookAccess	<ul style="list-style-type: none"> • Edit
DefaultTerritoryAccountAccess	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Default access level for price books. The possible values are listed, followed by the user interface labels in parentheses:</p> <ul style="list-style-type: none"> • None (No access) • Read (Read only) • ReadSelect (Use)
DefaultTerritoryCaseAccess	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Default access level for cases associated with accounts in territories. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All
DefaultTerritoryContactAccess	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Default access level for contacts associated with accounts in territories. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit

Field	Details
	<p>Description Default access level for contacts associated with accounts in territories. The possible values are:</p> <ul style="list-style-type: none"> • NoneRead • Edit <p> Note: When DefaultContactAccess is set to "Controlled by Parent" you can't update this field.</p>
DefaultTerritoryOppAccess	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Default access level for opportunities in territories. Valid values: <ul style="list-style-type: none"> • NoneRead • Edit </p>
Division	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description The name of the division for this organization. This field is not related to the Division object.</p>
Fax	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Fax number. Limit: 40 characters.</p>
FiscalYearStartMonth	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number that corresponds to the month that this organization's fiscal year starts.</p>

Field	Details
HomepageHtml	<p>Type textarea</p> <p>Properties Nillable, Update</p> <p>Description The Home tab custom links and company message for this organization.</p>
InstanceName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read-only. The name of the instance. Available in API version 31.0 or later.</p>
IsSandbox	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. Indicates whether the current organization is a sandbox (<code>true</code>) or production (<code>false</code>) instance. Available in API version 31.0 or later.</p>
LanguageLocaleKey	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The same as <code>Language</code>, the two-to-five character code which represents the language and locale ISO code. This controls the language for labels displayed in an application.</p>
LastWebToCaseDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description The last date that a web form submission was converted to a case.</p>

Field	Details
LastWebToLeadDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description The last date that a web form submission was converted to a lead.</p>
Latitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>Longitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
Longitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>Latitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
MaxActionsPerRule	<p>Type int</p> <p>Properties Filter, Nillable</p> <p>Description Maximum number of actions per workflow, assignment, escalation, and auto-response rules. This field is unavailable in version 15.0 and later.</p>
MaxRulesPerEntity	<p>Type int</p> <p>Properties Filter, Nillable</p>

Field	Details
	<p>Description Maximum number of rules per object, inclusive of workflow, assignment, escalation, and auto-response rules. This field is unavailable in version 15.0 and later.</p>
MonthlyPageViewsEntitlement	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of page views allowed for the current calendar month for the sites in your organization. To access this field, Force.com Sites must be enabled for your organization. This field is generally available in API versions 18.0 and later.</p>
MonthlyPageViewsUsed	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of page views used in the current calendar month for the sites in your organization. To access this field, Force.com Sites must be enabled for your organization. This field is generally available in API versions 18.0 and later.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The name of the organization.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p>

Field	Details
	<ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
<code>OrganizationType</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Edition of the organization, for example Enterprise Edition or Unlimited Edition.</p>
<code>Phone</code>	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Phone number for the organization.</p>
<code>PostalCode</code>	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Postal code for the address of the organization. Limit: 20 characters.</p>
<code>PreferencesEventScheduler</code>	<p>Type boolean</p> <p>Properties Update</p> <p>Description Indicates whether opportunities require products (<code>true</code>) or not (<code>false</code>).</p>

Field	Details
PreferencesRequireOpportunityProducts	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether opportunities require products (<code>true</code>) or not (<code>false</code>).</p>
PreferencesS1BrowserEnabled	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the Salesforce1 mobile browser app is enabled for all users in your organization (<code>true</code>) or is disabled for all users (<code>false</code>). This field is available in API version 29.0 or later.</p>
PreferencesTerminateOldestSession	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the oldest login session is automatically closed when a policy specifying the maximum number of sessions is triggered. This field is available in API version 35.0 or later.</p>
PreferencesTransactionSecurityPolicy	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description Indicates whether the Transaction Security feature has been enabled. This field is available in API version 35.0 or later.</p>
PrimaryContact	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description Name of the primary contact for the organization. Limit: 80 characters.</p>
ReceivesAdminInfoEmails	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the organization receives administrator emails (<code>true</code>) or not (<code>false</code>).</p>
ReceivesInfoEmails	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the organization receives informational email from Salesforce (<code>true</code>) or not (<code>false</code>).</p>
SelfServiceCasePlural	<p>Type string</p> <p>Properties Filter, Nullable, Update</p> <p>Description The plural version of the term used to represent the Case object in the Self-Service portal.</p>
SelfServiceCaseSingle	<p>Type string</p> <p>Properties Filter, Nullable, Update</p> <p>Description The singular version of the term used to represent the Case object in the Self-Service portal.</p>
SelfServiceCaseSubmitRecordTypeId	<p>Type reference</p> <p>Properties Filter, Nullable, Update</p>

Field	Details
	<p>Description The ID of the record type associated with a case submitted via the Self-Service portal.</p>
SelfServiceDefaultCaseOrigin	<p>Type string</p> <p>Properties Filter, Nullable, Update</p> <p>Description The default origin of a case submitted via the Self-Service portal.</p>
SelfServiceEmailSenderAddress	<p>Type email</p> <p>Properties Filter, Nullable, Update</p> <p>Description The Self-Service email address from which new Self-Service user and password email messages are sent, such as support@acme.com.</p>
SelfServiceEmailSenderName	<p>Type string</p> <p>Properties Filter, Nullable, Update</p> <p>Description The name associated with the email address in the SelfServiceEmailSenderAddress field, such as Acme Customer Support.</p>
SelfServiceEmailUserOnCaseCreationTemplateId	<p>Type reference</p> <p>Properties Filter, Nullable, Update</p> <p>Description The ID of the email template used when email is sent to a Self-Service user when he or she creates a case.</p>
SelfServiceEnabledForResponseRules	<p>Type boolean</p> <p>Properties Filter, Nullable, Update</p>

Field	Details
	Description
	Indicates whether the Self-Service portal is enabled for auto-response rules (<code>true</code>) or not (<code>false</code>).
SelfServiceFeatureConfig	Type int
	Properties
	Filter, Nillable, Update
	Description
	An integer representing the active Self-Service feature configuration for this organization.
SelfServiceLogoutUrl	Type url
	Properties
	Filter, Nillable, Update
	Description
	The Web page that displays when a Self-Service user logs out of the Self-Service portal.
SelfServiceMaxNumSuggestions	Type int
	Properties
	Filter, Nillable, Update
	Description
	The maximum number of suggested solutions allowed for a Self-Service case.
SelfServiceNewCommentCheckedByDefault	Type boolean
	Properties
	Defaulted on create, Filter, Update
	Description
	If <code>true</code> , When a customer notification is automatically sent when a new comment is added to a case.
SelfServiceNewCommentTemplateId	Type reference
	Properties
	Filter, Nillable, Update

Field	Details
SelfServiceNewPassTemplateId	<p>Description The ID of the email template used to send a notification to Self-Service users when a public comment is added to one of their cases.</p>
SelfServiceNewUserTemplateId	<p>Type reference</p> <p>Properties Filter, Nillable, Update</p> <p>Description The ID of the email template used when new passwords are generated for Self-Service users.</p>
SelfServicePageHeight	<p>Type reference</p> <p>Properties Filter, Nillable, Update</p> <p>Description The ID of the email template used when new Self-Service users are enabled.</p>
SelfServicePageWidth	<p>Type int</p> <p>Properties Filter, Nillable, Update</p> <p>Description The maximum height in pixels of Self-Service pages.</p>
SelfServiceSelfClosedCaseStatus	<p>Type int</p> <p>Properties Filter, Nillable, Update</p> <p>Description The maximum width in pixels of Self-Service pages.</p>
	<p>Type picklist</p> <p>Properties Filter, Nillable, Update</p> <p>Description The default status for cases closed by Self-Service users.</p>

Field	Details
SelfServiceSolutionCategoryAvailable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Update</p> <p>Description Indicates whether solution categories are available in the Self-Service portal (<code>true</code>) or not (<code>false</code>).</p>
SelfServiceSolutionCategoryStartNodeId	<p>Type reference</p> <p>Properties Filter, Nillable, Update</p> <p>Description The ID of the top-level category in the Self-Service portal.</p>
SelfServiceSolutionPlural	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description The plural version of the term used to represent the Solution object in the Self-Service portal.</p>
SelfServiceSolutionSingle	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description The singular version of the term used to represent the Solution object in the Self-Service portal.</p>
SelfServiceStyleSheetUrl	<p>Type url</p> <p>Properties Filter, Nillable, Update</p> <p>Description The public URL of your organization's Self-Service portal stylesheet.</p>
SelfServiceWelcomePageConfig	<p>Type int</p> <p>Properties Filter, Nillable, Update</p>

Field	Details
	<p>Description Integer that represents the welcome page configuration for the Self-Service portal.</p>
SelfServiceWelcomeText	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description The custom welcome message displayed at the top of the Self-Service home page when Self-Service users log in. Limit: 32,000 characters.</p>
SignupCountryIsoCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ISO country code specified by the user for a sign-up request.</p>
State	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description State of the address of the organization. Limit: 80 characters.</p>
StateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the organization's address.</p>
Street	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Street address for the organization. Limit: 255 characters.</p>

Field	Details
TrialExpirationDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date that this organization's trial license expires.
UiSkin	Type picklist
	Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The user interface theme selected for the organization.
UsesStartDateAsFiscalYearName	Type boolean
	Properties Defaulted on create, Filter, Group, Sort
	Description Indicates whether the calendar year when the fiscal year begins is referred to as the year of the company's fiscal year (<code>true</code>) or not (<code>false</code>). For example, if the fiscal year begins in February 2006, a <code>true</code> value means the fiscal year is FY2006, and a <code>false</code> value means the fiscal year is FY2007.
UsesWebToCase	Type boolean
	Properties Filter, Nillable, Update
	Description Indicates whether this organization can use Web-to-Case (<code>true</code>) or not (<code>false</code>).
UsesWebToLead	Type boolean
	Properties Filter, Nillable, Update
	Description Indicates whether this organization can use Web-to-Lead (<code>true</code>) or not (<code>false</code>).

Field	Details
WebToCaseAssignedEmailTemplateId	Type reference Properties Filter, Nillable, Update Description The ID of the email template used when a new case is assigned to a user via Web-to-Case.
WebToCaseCreatedEmailTemplateId	Type reference Properties Filter, Nillable, Update Description The ID of the email template used when a new case is created via Web-to-Case.
WebToCaseDefaultCreatorId	Type reference Properties Filter, Nillable, Update Description The ID of the user specified as the default creator of cases created via Web-to-Case.
WebToCaseDefaultOrigin	Type string Properties Filter, Group, Nillable, Sort, Update Description The default value for the Case Origin field on cases submitted via Web-to-Case. Limit: 40 characters.

Usage

Query this object to obtain information about an organization's settings. Only one organization object exists per organization.

SEE ALSO:

[Object Basics](#)

OrgWideEmailAddress

Represents an organization-wide email address for user profiles.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Address	<p>Type email</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The organization-wide email address.</p>
IsAllowAllProfiles	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, any user profile in your organization can use this object. If <code>false</code>, only specified user profiles can use this object when sending email. If you do not have the appropriate user profile, you can't use this object.</p>
DisplayName	<p>Type string</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The name that is used to identify the sender of the email.</p>

Usage

This object represents an email alias for user profiles. You can pass in the ID to an OrgWideEmailAddress record when calling `sendEmail()` for a SingleEmailMessage.

OwnedContentDocument

Represents a file owned by a user. This object is available in version 30.0 and later.

Supported Calls

`describeSObjects()`

Fields

Field Name	Details
ContentDocumentId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the document.</p>
ContentSize	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Size of the document in bytes.</p>
ContentUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL for links and Google Docs. This field is set only for links and Google Docs, and is one of the fields that determine the <code>FileType</code>. This field is available in API version 31.0 and later.</p>
ExternalDataSourceName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.</p>

Field Name	Details
ExternalDataSourceType	<p>This field is available in API version 32.0 and later.</p>
FileExtension	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Type of external data source in which the document is stored. This field is set only for external documents that are connected to Salesforce.</p> <p>This field is available in API version 35.0 and later.</p>
FileType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description File extension of the document.</p> <p>This field is available in API version 31.0 and later.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the owner of the document.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description Title of the document.</p>

PackageLicense

Represents a license for an installed managed package. This object is available in API version 31.0 and later.

Supported Calls

`query()`, `retrieve()`

Fields

Field Name	Details
AllowedLicenses	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of users allowed to use the package.</p>
ExpirationDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time when the package license expires.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The namespace prefix associated with the package.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description The status of the license. Possible values are: Active, Expired, Free, and Trial.</p>
UsedLicenses	<p>Type int</p>

Field Name	Details
Properties	Create, Filter, Nillable, Sort
Description	The number of users who have a license to the package.

Usage

Use this object to determine the number of licenses allowed and in use for a managed package installed in your organization.

The following example demonstrates the use of the API to manage licenses for a package. The example defines an Apex class that does the following.

- Retrieves the PackageLicense record for the specified package (identified by its namespace prefix).
- Defines a function that returns a list of all users with the specified profile.
- Creates a UserPackageLicense record for each user with that profile, which has the effect of assigning a license for the package to all users with that profile.
- Returns an error message if the number of users exceeds the number of available licenses.

```
public class AssignPackageLicense {

    static String PACKAGE_NAMESPACE_PREFIX = 'acme_101';
    static String PROFILE_ID = '00e0xx000000jz1SAAQ';
    public static String exceptionText {get; set;}

    public AssignPackageLicense() {
        exceptionText = 'Initialized';
    }

    static List<User> getUsersWithProfile(){
        String userQuery = 'SELECT Id FROM User WHERE ProfileId = :PROFILE_ID';
        List<User> matchingUsers = new List<User>();
        matchingUsers = [SELECT Id FROM User WHERE ProfileId = :PROFILE_ID];
        return matchingUsers;
    }

    public static void assignLicenseByProfile() {
        //find the PackageLicense Id
        PackageLicense pl = [SELECT Id, NamespacePrefix, AllowedLicenses, UsedLicenses,
            ExpirationDate, Status FROM PackageLicense WHERE
            NamespacePrefix = :PACKAGE_NAMESPACE_PREFIX];
        System.assert(pl != null, 'PackageLicense cannot be null.');
        List<User> usersToAssignLicenses = getUsersWithProfile();
        List<UserPackageLicense> firstUPLs = new List<UserPackageLicense>();

        //create a new UserPackageLicense record for each user with the specified profile

        for (Integer i = 0; i < usersToAssignLicenses.size(); i++) {
            UserPackageLicense upl = new UserPackageLicense();
            upl.PackageLicenseId = pl.Id;
        }
    }
}
```

```
        upl.UserId = usersToAssignLicenses[i].Id;
        firstUPLs.add(upl);
    }

    try {
        //bulk insert
        insert(firstUPLs);
    } catch(DmlException e) {
        for (Integer i = 0; i < e.getNumDml(); i++) {
            // process exception here
            System.debug(e.getDmlMessage(i));
            String status = e.getDmlStatusCode(i);
            System.debug(status + ' ' + e.getDmlMessage(i));
            if(status.equals('LICENSE_LIMIT_EXCEEDED')){
                exceptionText = 'You tried to assign more licenses than available. '
                +' You tried to create '+ firstUPLs.size()+' licenses but only have '
                + (pl.AllowedLicenses - pl.UsedLicenses) + ' licenses free.';
                System.debug(exceptionText);
            }
        }
    }
}
```

PackagePushError

Represents an error encountered during a push request. The number of PackagePushError records created depends on the number of push jobs in the request that result in an error.

Supported Calls

`describeSOBJECTS(), query(), retrieve()`

Fields

Field Name	Details
ErrorDetails	Type string Properties Nullable, Sort
ErrorMessage	Description Explanation of the error.

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The error code that appears in the API.</p>
ErrorSeverity	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Valid values are:</p> <ul style="list-style-type: none">• Error• Warning
ErrorTitle	<p>Type string</p> <p>Properties Nillable, Sort</p> <p>Description The error message title that appears in the API.</p>
ErrorType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Valid values are:</p> <ul style="list-style-type: none">• ApexTestFailure• DeployError• FeatureMissing• IneligibleUpgrade• LimitExceeded• LockingFailure• PACError• UnclassifiedError
PackagePushJobId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description
	Required. The parent push job record ID.

Usage

Suppose that your push upgrade request wasn't successful due to some of its jobs failing. Let's write some code to find out what those errors were.

This code sample uses the Force.com Web Services Connector (WSC).

```
// Retrieves all PackagePushError objects associated with the PackagePushJob with the given
// ID
final String PACKAGE_PUSH_ERROR_QUERY = "Select ErrorMessage, ErrorDetails, ErrorTitle,"
+ " ErrorSeverity, ErrorType from PackagePushError where PackagePushJobId = '%s'";

// job is a PackagePushJob instance
QueryResult queryResult = conn.query(String.format(PACKAGE_PUSH_ERROR_QUERY, job.getId()));

StringBuilder errorMessages = new StringBuilder();
errorMessages.append("Errors for PackagePushJob [").append(job.getId()).append("]:")

.append("\n");

// There can be multiple PackagePushErrors for a given PackagePushJob
for(SObject r : queryResult.getRecords()) {
    PackagePushError e = (PackagePushError) r;
    errorMessages.append("Title: ").append(e.getErrorTitle()).append("\n");
    errorMessages.append("Severity: ").append(e.getErrorSeverity()).append("\n");
    errorMessages.append("Type: ").append(e.getErrorType()).append("\n");
    errorMessages.append("Message: ").append(e.getErrorMessage()).append("\n");
    errorMessages.append("Details: ").append(e.getErrorDetails()).append("\n");
    errorMessages.append("\n");
}

String errors errorMessages.toString();
```

PackagePushJob

Represents an individual push job for upgrading a package in an org from one version to another version. There can be multiple push jobs created for one push request. For example, if you want to upgrade five orgs as part of one push, you have one PackagePushRequest record and five PackagePushJob records.

Supported Calls

`create(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
PackagePushRequestId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Required. The ID of the parent push request record which must have been created.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description The status of the job. Valid values are:</p> <ul style="list-style-type: none"> • Cancelled • Created (default) • Failed • In Progress • Pending • Succeeded <p>Don't specify this value when you create the push job. The default value of <code>Created</code> is used.</p>
SubscriberOrganizationKey	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Required. The organization key of the org where the package is upgraded. This references <code>orgKey</code> in <code>PackageSubscriber</code>.</p>

Usage

Suppose that you want to push version 3.4.6 of your package to all orgs. You've already identified the orgs eligible for the upgrade by using `MetadataPackageVersion` and created the push request using `PackagePushRequest`. Now let's write some code to create a push job for each eligible org.

This code sample uses the Force.com Web Services Connector (WSC).

```
PackageSubscriber[] subscribers = new PackageSubscriber[];
```

```
// ... populate eligible and desired subscribers

// Create the PackagePushJob array
PackagePushJob[] jobs = new PackagePushJob[subscribers.length];

for (int i = 0; i < subscribers.length; i++) {
    // create a job for each subscriber...
    PackagePushJob job = new PackagePushJob();
    // ... associate it to the PackagePushRequest ppr...
    job.setPackagePushRequestId(ppr.getId());
    // ... and add the orgKey
    job.setSubscriberOrganizationKey(subscribers[i].getOrgKey());
    jobs[i] = job;
}

// Save the jobs
SaveResult[] saveResults = conn.create(jobs);

// Add the newly generated id's to the PackagePushJob objects
for (int i = 0; i < saveResults.length; i++) {
    if (saveResults[i].isSuccess()) {
        jobs[i].setId(saveResults[i].getId());
    }
}
}
```

Or, if you're using REST API, submit a POST request to the PackagePushJob sObject endpoint, as in the following example. SOAP API is also supported. This example returns the push job ID (starting with 0DX) that is required to query the status of the job.

```
POST
/services/data/v38.0/sobjects/packagepushjob/
{
    "PackagePushRequestId" : "0DV...",
    "SubscriberOrganizationKey" : "00DR00..."
}
```

Checking the Status of a Push Job

To check the job status, simply query the `Status` field. For example:

```
SELECT Id, Status FROM PackagePushJob WHERE PackagePushRequestId ='0DV...'
```

Here's an example in Java.

```
// Finds the status of the PackagePushJob with the given id
String PACKAGE_PUSH_JOB_STATUS_QUERY = "Select status from PackagePushJob where Id = '%s'";

// job is a PackagePushJob instance
QueryResult queryResult = conn.query(String.format(PACKAGE_PUSH_JOB_STATUS_QUERY,
job.getId()));

// extract the status from the QueryResult
String status = ((PackagePushJob) queryResult.getRecords()[0]).getStatus();

// optionally, update the PackagePushJob instance with the latest status
job.setStatus(status);
```

You can also continuously poll the job status until the job is done. The following Java example polls the status every 10 seconds.

```
// The set of states that indicate a PackagePushJob has completed
final Set<String> TERMINAL_STATES = new HashSet<>();
TERMINAL_STATES.add("Succeeded");
TERMINAL_STATES.add("Failed");
TERMINAL_STATES.add("Canceled");

String status = queryJobStatus(job); // this method returns the status as retrieved in the
// previous code sample

// If the status is not one of the completed statuses...
while (!TERMINAL_STATES.contains(status)) {
    Thread.sleep(10 * 1000); // ... wait 10 seconds and try again
    status = queryJobStatus(job);
}
```

PackagePushRequest

Represents the push request for upgrading a package in one or many orgs from one version to another version.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
PackageVersionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Required. The non-beta, non-deprecated package version that the package is being upgraded to.</p>
ScheduledStartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date and time (UTC) at which the push request is processed, in ISO 8601 format. Set this value to the earliest time that you want Salesforce to attempt to start the push. As a best practice, schedule pushes at off-peak hours like 1:00 AM Saturday. If you don't specify a value, the push starts when the package push request's Status is set to Pending.</p>

Field Name	Details
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The status of the push. Valid values are:</p> <ul style="list-style-type: none"> • Cancelled • Created (default) • Failed • In Progress • Pending • Succeeded <p>Don't specify this value when you create the push request. The default value of <code>Created</code> is used. Later, you need to change the status to <code>Pending</code> to schedule the push upgrade.</p>

Usage

Suppose that you want to push version 3.4.6 of your package to all orgs. You've already identified the orgs eligible for the upgrade by using `MetadataPackageVersion`. Now let's write some code to create a push request, which holds a push job for each eligible org.

This code sample uses the Force.com Web Services Connector (WSC).

```
// Create a new PackagePushRequest for the versionId to upgrade to
// (for example, versionId is the "04t..." id of version
// 3.4.6 of the package
PackagePushRequest ppr = new PackagePushRequest();
ppr.setPackageVersionId(versionId);

// Optionally, set the start time of the PackagePushRequest to schedule it to begin
// automatically; scheduledStartTime is a java.util.Calendar instance
ppr.setScheduledStartTime(scheduledStartTime);

// Save the PackagePushRequest
SaveResult[] saveResults = conn.create(new SObject[] { ppr });
if (saveResults[0].isSuccess()) {
    // Add the newly generated Id to the object
    ppr.setId(saveResults[0].getId());
} else {
    for (Error error : saveResults[0].getErrors()) {
        System.out.println(error.getMessage());
    }
}
```

Or, if you're using REST API, submit a POST request to the `PackagePushRequest` sObject endpoint, as in the following example. SOAP API is also supported.

This example returns the push request ID (starting with 0DV) that's required to create push jobs.

```
POST
/services/data/v38.0/sobjects/packagepushrequest/
{
    "PackageVersionId" : "04t...",
    "ScheduledStartTime" : "2016-08-24T21:00:00"
}
```

As your next step, create a push job for each eligible subscriber you want to upgrade using PackagePushJob.

Scheduling the Push Upgrade

To signal that the push upgrade is ready to be processed, change the status of the push request to Pending. If you didn't set a ScheduledStartTime, the push upgrade starts immediately after you change the status.

See the following Java example.

```
// ppr is the PackagePushRequest instance
ppr.setStatus("Pending");
conn.update(new SObject[] { ppr });
```

If you're using REST API, submit a PATCH request to the PackagePushRequest sObject endpoint, as in the following example. SOAP API is also supported.

```
PATCH
/services/data/v38.0/sobjects/packagepushrequest/0DV...
{
    "Status" : "Pending"
}
```

Checking the Status of a Push Request

The PackagePushRequest status is Succeeded if all its associated jobs are successful; it's Failed if at least one job failed.

```
// Finds the status of the PackagePushRequest for a given Id
final String PACKAGE_PUSH_REQUEST_STATUS_QUERY = "Select status from PackagePushRequest"
+
" where Id = '%s'";

// ppr is a PackagePushRequest instance
QueryResult queryResult = conn.query(String.format(PACKAGE_PUSH_REQUEST_STATUS_QUERY,
ppr.getId()));

// extract the status from the QueryResult
String status = ((PackagePushRequest) queryResult.getRecords()[0]).getStatus();

// optionally, update the PackagePushRequest instance with the latest status
ppr.setStatus(status);
```

You can also check the status of a job by querying the PackagePushJob's Status field.

PackageSubscriber

Represents an installation of a package in an org. This object contains installation information for managed packages developed in the org you're logged in to.

One record is created per installation. For example, if 5 orgs installed 2 packages, 10 records are created.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
InstalledStatus	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description If the package is installed in the org, the value is i.</p>
InstanceName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The instance that hosts the subscriber org.</p>
MetadataPackageVersionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 18-character package version ID starting with 04t.</p>
OrgKey	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 18-character ID that represents the Salesforce org.</p>
OrgName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the org where the package is installed.</p>

Field Name	Details
OrgStatus	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Valid values are:</p> <ul style="list-style-type: none"> • Active • Demo • Free • Inactive • Trial <p>Orgs with an <code>OrgStatus</code> of <code>Inactive</code> can't receive push upgrades.</p>
OrgType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Valid values are:</p> <ul style="list-style-type: none"> • Production • Sandbox

Usage

Here are examples of the types of API queries you can perform.

Query	String
Get all package subscriber orgs with a specific package ID	<code>SELECT Id, OrgKey, OrgStatus, OrgName, OrgType FROM PackageSubscriber WHERE MetadataPackageVersionId = '04t...'</code>
Get all package subscriber orgs that have an installed package created by the org you're logged in to	<code>SELECT Id, OrgKey, OrgStatus, OrgName, OrgType FROM PackageSubscriber WHERE InstalledStatus = 'i'</code>

Filter PackageSubscriber Objects by Instance

If you have packages with many subscribers, querying `PackageSubscriber` objects can take a while. To improve query performance, add filters to your `PackageSubscriber` queries, such as an `InstanceName` filter. `InstanceName` is a field that represents the instance that the subscriber org is hosted on.

- Get the org's managed package and the latest released version of the package.

```

/**
 * Get the MetadataPackage object corresponding to this org's managed package
 */
public MetadataPackage getMetadataPackage() throws ConnectionException {
    // retrieve the managed package, which won't have an empty namespace
    QueryResult result = conn.query("select id from MetadataPackage where namespaceprefix
    <> ''");

    return (MetadataPackage) result.getRecords()[0];
}

/**
 * Get the latest MetadataPackageVersion object of the given MetadataPackage
 */
public MetadataPackageVersion getLatestMetadataPackageVersion(MetadataPackage
metadataPackage)
throws ConnectionException {
    // get the latest released version of the given package
    String query = "Select id, ReleaseState, MajorVersion, MinorVersion, PatchVersion,
MetadataPackageId"
        + " From MetadataPackageVersion"
        + " Where MetadataPackageId = '%s' and ReleaseState = 'Released'"
        + " Order by majorversion desc, minorversion desc, patchversion desc";

    QueryResult result = conn.query(String.format(query, metadataPackage.getId()));

    return (MetadataPackageVersion) result.getRecords()[0];
}

```

- Get eligible subscribers. The following query strings and methods are modified to allow querying for PackageSubscribers filtered by an instance.

```

static final String PACKAGE_SUBSCRIBER_ORG_KEY_QUERY = "Select OrgKey from
PackageSubscribers where OrgStatus = 'Active'"
    + " and InstalledStatus = 'I'"
    + " and InstanceName = '%s"'; // placeholder for instance values

static final String METADATA_PACKAGE_VERSION_QUERY = "Select Id, Name, ReleaseState,
(%s) from MetadataPackageVersion"
    + " where MetadataPackageId = '%s' AND ReleaseState = 'Released'"
    + " AND (MajorVersion < %s OR (MajorVersion = %s and MinorVersion < %s))"
    + " OR (MajorVersion = %s and MinorVersion = %s and PatchVersion < %s))";

/**
 * Get all PackageSubscribers on the given instance that are eligible to upgrade to the
given
 * MetadataPackageVersion
 */
public PackageSubscriber[] getEligibleSubscriberIds(MetadataPackageVersion version,
String instanceName) throws ConnectionException {
    String allPackageId = version.getMetadataPackageId();
    Integer major = version.getMajorVersion();
    Integer minor = version.getMinorVersion();
}

```

```

    Integer patch = version.getPatchVersion();

    return getEligibleSubscriberIds(major, minor, patch, allPackageId, instanceName);
}

public PackageSubscriber[] getEligibleSubscriberIds(Integer major, Integer minor, Integer
patch, String packageId, String instanceName) throws ConnectionException {
    String subscriberQuery = String.format(PACKAGE_SUBSCRIBER_ORG_KEY_QUERY, instanceName);
    QueryResult results = conn.query(String.format(METADATA_PACKAGE_VERSION_QUERY,
    subscriberQuery, packageId, major, major, minor, major, minor, patch));

    return Arrays.stream(results.getRecords()) .map(MetadataPackageVersion.class::cast)
        .filter(mpv -> mpv.getPackageSubscribers() != null)
        .flatMap(mpv -> Arrays.stream(mpv.getPackageSubscribers().getRecords()))
        .map(PackageSubscriber.class::cast)
        .toArray(PackageSubscriber[]::new);
}

```

- 3.** Put it all together. The following code sample shows how to use the previous methods to modify the workflow to perform package pushes by instance.

```

String[] instances = { "NA4" }; // Here we list the instances we would like to push to
MetadataPackage metadataPackage = api.getMetadataPackage();
MetadataPackageVersion version = api.getLatestMetadataPackageVersion(metadataPackage);

// do pushes by instance to avoid API timeouts retrieving PackageSubscribers
for (String instanceName : instances) {
    PackageSubscriber[] eligibleSubscribers = api.getEligibleSubscriberIds(version,
    instanceName);

    // ... proceed with creating PushRequests and PushJobs as before
}

```

Partner

Represents a partner relationship between two Account records or between an Opportunity and an Account.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

- You must have the “View All Data” permission to access this object via the API. All of the Partner fields are accessible in the `describeSObjects()` and `query()` calls. You cannot `update()` or `upsert()` partners via the API.
- Customer Portal users can’t access this object.

Fields

Field	Details
AccountFromId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description Required if OpportunityId is null. ID of the main Account in a partner relationship between two accounts. Specifying this field when creating a Partner record creates two AccountPartner objects, one for each direction of the relationship. If you specify the OpportunityId field, you can't specify this field as well.</p>
AccountToId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Partner Account related to either an opportunity or an account. You must specify this field when creating an opportunity Partner or an Account Partner.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsPrimary	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Valid for Opportunity Partners only. Indicates that the account is the primary partner for the opportunity. Only one account can be marked as primary for an opportunity. If you set this field to 1 (<code>true</code>) upon insert of a new opportunity partner, any other primary partners for that opportunity will automatically have this field set to 0 (<code>false</code>). Label is Primary.</p>
OpportunityId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Required if <code>AccountFromId</code> is null. ID of the Opportunity in a partner relationship between an Account and an Opportunity. Specifying this field when creating a record creates an <code>OpportunityPartner</code>. If you specify the <code>AccountFromId</code> field, you can't specify this field as well.</p>
<code>Role</code>	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort,</p> <p>Description UserRole that the account has towards the related opportunity or account, such as consultant or distributor.</p>

Roles

In the Salesforce user interface, system administrators can set up the valid role values and their corresponding reverse role values in the `PartnerRole` object. Each account in the relationship is assigned a `Role` (such as `Consultant` or `Distributor`) designating that account's role towards the related account or opportunity.

Creating an Account-Opportunity Partner Relationship

When you create a partner relationship between an account and an opportunity (when you create a `Partner` object and specify the `OpportunityId` field), the API automatically creates an `OpportunityPartner` with the corresponding values:

- The value of the Partner field `AccountToId` maps to the value of the `OpportunityPartner` field `AccountToId`.
- The values of the `OpportunityId`, `Role`, and `IsPrimary` fields in both objects are the same.
- If you set the `IsPrimary` value to 1 (`true`) upon insert of a new `OpportunityPartner`, any other existing primary partners for that opportunity will automatically have the `IsPrimary` value set to 0 (`false`).

This mapping allows the API to manage the objects and their relationship efficiently.

Creating an Account-Account Partner Relationship

When you create a partner relationship between two accounts (when you create a `Partner` object and specify the `AccountFromId`), the API automatically creates two `AccountPartner` objects, one for the forward relationship and one for the reverse. For example, if you create a `Partner` object with "Acme, Inc." as the `AccountFromId` and "Acme Consulting" as the `AccountToId`, the API automatically creates two `AccountPartner` objects:

- The forward relationship `AccountPartner` with "Acme, Inc." as the `AccountFromId` and "Acme Consulting" as the `AccountToId`.
- The reverse relationship `AccountPartner` with "Acme Consulting" as the `AccountFromId` and "Acme, Inc." as the `AccountToId`.
- The value of the `Role` field in the reverse relationship `AccountPartner` is set to the `PartnerRole` object `ReverseRole` value associated with the value of the `Role` field in the forward relationship `AccountPartner`.

This mapping allows the API to manage the objects and their relationship efficiently.

SEE ALSO:

[AccountPartner](#)
[OpportunityPartner](#)
[UserRole](#)
[PartnerRole](#)

PartnerNetworkConnection

Represents a Salesforce to Salesforce connection between Salesforce organizations.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the Account associated with this connection.</p>
ConnectionName	<p>Type string</p> <p>Properties Filter, idLookup, Sort</p> <p>Description A descriptive name for the connection. Limit: 295 characters.</p>
ConnectionStatus	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The status of the Salesforce to Salesforce connection. The picklist includes the following values:<ul style="list-style-type: none">• Sent</p>

Field	Details
	<ul style="list-style-type: none"> • Received • Pending • Accepted • Rejected • Inactive • Disconnecting • ConnectionSuspended • SubscribeInProgress • UsersInitialSync • BulkSyncMetadata
ConnectionType	Type
	picklist
	Properties
	Defaulted on create, Filter, Group, Restricted picklist, Sort
	Description
	The type of Salesforce to Salesforce connection. The picklist includes the following values:
	<ul style="list-style-type: none"> • Standard • Replication
	This field is available in API version 30.0 and later.
ContactId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	ID of the Contact associated with this connection.
IsSyncAuditFields	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Specifies whether audit fields are synced between the primary and secondary organization in a replication connection. This field is available in API version 32.0 and later, and is only accessible in Salesforce organizations where Organization Sync is enabled.
IsSyncMetadata	Type
	boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Specifies whether supported types of metadata are synced from the primary to the secondary organization in a replication connection. This field is available in API version 33.0 and later, and is only accessible in Salesforce organizations where Organization Sync is enabled.</p>
IsSyncUsers	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Specifies whether users with standard Salesforce user licenses are synced between the primary and secondary organization in a replication connection. This field is available in API version 35.0 and later, and is only accessible in Salesforce organizations where Organization Sync is enabled.</p>
PrimaryContactId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the User associated with this connection.</p>
ReplicationRole	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The role of this Salesforce organization in the replication connection. The picklist includes the following values:</p> <ul style="list-style-type: none"> • Primary • Secondary <p>This field is available in API version 30.0 and later, and is only accessible in Salesforce organizations where Organization Sync is enabled.</p>
ResponseDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description The date and time that the connection was accepted or rejected.</p>

Usage

Represents Salesforce to Salesforce standard and replication connections. This object is referenced by all objects that have been shared with other organizations, enabling you to determine which connections shared a record with you. If the organization does not have Salesforce to Salesforce enabled, the PartnerNetworkConnection object is not available, and you can't access it via the API.

SEE ALSO:

[PartnerNetworkRecordConnection](#)

PartnerNetworkRecordConnection

Represents a record shared between Salesforce organizations using Salesforce to Salesforce.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`

Fields

Field	Details
ConnectionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the connection a record is shared with.</p>
EndDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date that sharing of the record was stopped.</p>
LocalRecordId	<p>Type reference</p>

Field	Details
	Properties
	Create, Filter, Group, Sort
	Description
	Required. ID of the shared record.
ParentRecordId	Type
	reference
	Properties
	Create, Filter, Group, Nillable, Sort
	Description
	ID of the parent record of the shared record.
PartnerRecordId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	ID of the shared record in the connection's organization.
RelatedRecords	Type
	string
	Properties
	Create, Filter, Nillable, Sort
	Description
	A comma-separated list of API names for child records to be shared with a parent record.
SendClosedTasks	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort
	Description
	Forwards closed tasks related to the shared record.
SendEmails	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort
	Description
	Sends an email notifying the connection's representative that you have forwarded the record to them. Only new recipients of a record will receive a notification email.

Field	Details
SendOpenTasks	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Forwards open tasks related to the shared record.</p>
StartDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date that the shared record was accepted.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The status of the shared record. One of the following values:</p> <ul style="list-style-type: none"> • Active (received) • Active (sent) • Connected • Inactive • Inactive (converted) • Inactive (deleted) • Pending (sent)

Usage

When you create a PartnerNetworkRecordConnection, you forward a record to a connection.

-  **Note:** Attempting to forward a record from an object to which the connection is not subscribed results in an `Invalid Partner Network Status` error.

Don't forward a record to the connection that originally shared it. Doing so causes errors when Apex triggers run.

When you delete a PartnerNetworkRecordConnection, you stop sharing a record with a connection.

- To share a record, use the following fields: `LocalRecordID` and `ConnectionID`
- To share a child of a parent record, use the following fields: `LocalRecordID`, `ConnectionID`, and `ParentRecordID`
- To share a child of a parent record and its child records, use the following fields: `LocalRecordID`, `ConnectionID`, `ParentRecordID`, and `RelatedRecords`

If the organization does not have Salesforce to Salesforce enabled, the PartnerNetworkRecordConnection object is not available, and you can't access it using the API.

Sample Code—Apex

The following example shows how to forward a record.

```
List<PartnerNetworkConnection> connMap = new List<PartnerNetworkConnection>(
    [select Id, ConnectionStatus, ConnectionName from PartnerNetworkConnection
        where ConnectionStatus = 'Accepted']
);
for(PartnerNetworkConnection network : connMap) {
    PartnerNetworkRecordConnection newrecord = new PartnerNetworkRecordConnection();

    newrecord.ConnectionId = network.Id;
    newrecord.LocalRecordId = accountId;
    newrecord.RelatedRecords = 'Contact,Opportunity,Orders__c';
    newrecord.SendClosedTasks = true;
    newrecord.SendOpenTasks = true;
    newrecord.SendEmails = true;

    insert newrecord;
}
```

The following example shows how to stop sharing a record.

```
List<PartnerNetworkRecordConnection> recordConns = new List<PartnerNetworkRecordConnection>(
    [select Id, Status, ConnectionId, LocalRecordId from PartnerNetworkRecordConnection
        where LocalRecordId in :accounts]
);

for(PartnerNetworkRecordConnection recordConn : recordConns) {
    if(recordConn.Status.equalsIgnoreCase('Sent')){ //account is connected - outbound

        delete recordConn;
    }
}
```

SEE ALSO:

[PartnerNetworkConnection](#)

PartnerNetworkSyncLog

Represents the Org Sync Log tab in Salesforce, where Salesforce administrators can track the replication of record inserts and updates being performed in Organization Sync. The Connection Detail page for the replication connection also displays the Org Sync Log's twenty most recent entries, and provides a link to the log.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search()`

Special Access Rules

The Org Sync Log tab can only be added in organizations where Organization Sync has been enabled. To add the tab to the Salesforce user interface, users must also have the “Manage Connections” user permission.

Fields

Field Name	Details
ConnectionEvent	Type picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	The action being replicated to the partner organization, such as a record insertion.
ConnectionId	Type reference
	Properties
	Filter, Group, Sort
	Description
	The ID of the Salesforce to Salesforce replication connection in which the replication event succeeded or failed.
Description	Type textarea
	Properties
	Nillable
	Description
	A description of the replication event.
EntityType	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The type of record being inserted or updated.
Error	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The code used to describe the replication failure or success.

Field Name	Details
LocalRecord	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The record being inserted or updated.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description An item is added to the Organization Sync Log if it failed to be replicated to the linked organization. This picklist includes the following values:</p> <ul style="list-style-type: none"> • Failed: The replication continued to fail after multiple retries, and won't be retried further. • Resolved: The replication succeeded after retrying. • Retrying: Salesforce is retrying the replication. <p>This field is available in API version 35.0 and later.</p>

PartnerRole

Represents a role for an account Partner, such as consultant, supplier, and so on.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, NillableSort</p>

Field	Details
	<p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this partner role value. This display value is the internal label that does not get translated. Limit: 255 characters.</p>
ReverseRole	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the reverse role that corresponds to this partner role. For example, if the role is "subcontractor," then the reverse role might be "general contractor." In the user interface, assigning a partner role to an account creates a reverse partner relationship so that both accounts list the other as a partner.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the partner role picklist. These numbers are not guaranteed to be sequential, as some previous partner role values might have been deleted.</p>

Usage

This object represents a value in the partner role picklist. In the user interface, the partner role picklist provides additional information about the role of a Partner, such as their corresponding reverse role. Query this object to retrieve the set of values in the partner role picklist, and then use that information while processing PartnerRole records to determine more information about a given partner role. For example, the application could determine the reverse role of a given Partner Role value and the value of the ReverseRole property in the associated PartnerRole object.

SEE ALSO:

[Object Basics](#)

Period

Represents a fiscal period defined in FiscalYearSettings.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
EndDate	Type date Properties Filter, Group, Sort Description The last date of the fiscal period.
FiscalYearSettingsId	Type reference Properties Filter, Nillable, Group, Sort Description The parent record for this period.
FullyQualifiedLabel	Type string Properties Group, Nillable Description Represents the period's complete name in the UI. For example, "September FY 2016".
IsForecastPeriod	Type boolean Properties Defaulted on create, Filter, Group, Sort

Field	Details
	<p>Description Indicates whether the period is associated with customizable forecasts (<code>true</code>) or not (<code>false</code>).</p>
Number	<p>Type int</p> <p>Properties Filter, Nillable, Group, Sort</p> <p>Description If the labeling scheme of your fiscal year's quarters or months is numbered, this field indicates the relative number of the row.</p>
PeriodLabel	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the months in your fiscal year use custom names, then this field contains the appropriate name for rows of type Month.</p>
QuarterLabel	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Sort</p> <p>Description If the quarters in your fiscal year use custom names, then this field contains the appropriate name for rows of type Quarter.</p>
StartDate	<p>Type date</p> <p>Properties Filter, Group, Sort</p> <p>Description The first date of the fiscal period.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates whether the period is of type Month, Quarter, Week, or Year. Label is the field value.</p>

Usage

In API version 36.0 and earlier, querying the Period object yields no results. In API version 37.0 and later, a query returns period records.

SEE ALSO:

[FiscalYearSettings](#)

PermissionSet

Represents a set of permissions that's used to grant additional access to one or more users without changing their profile or reassigning profiles. This object is available in API version 22.0 and later.

You can use permission sets to grant access, but not to deny access.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Fields

Field Name	Details
Description	<p>Type string</p> <p>Properties Create, Filter, Nullable, Group, Sort, Update</p> <p>Description A description of the permission set. Limit: 255 characters.</p>
HasActivationRequired	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the permission set requires an associated active session (<code>true</code>) or not (<code>false</code>).</p>
IsOwnedByProfile	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the permission set is owned by a profile. Available in API version 25.0 and later.</p>

Field Name	Details
Label	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The permission set label, which corresponds to Label in the user interface. Limit: 80 characters.</p>
LicenseId	<p>Type ID</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of either the related PermissionSetLicense or UserLicense associated with this permission set. Available in API version 38.0 and later. Use this field instead of UserLicenseId, which is deprecated and only available up to API Version 37.0.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. Corresponds to API Name in the user interface. Limit: 80 characters.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix for a permission set that's been installed as part of a managed package. If the permission set isn't packaged or is part of an unmanaged package, this value is empty. Available in API version 23.0 and later.</p>
Permissions.PermissionName	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field Name	Details
	<p>Description One field for each permission. If true, users assigned to this permission set have the named permission. The number of fields varies depending on the permissions for the organization and license type.</p> <p> Tip: To get a list of available permissions in the SOAP API, use <code>describeSObjects()</code>.</p>
ProfileId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the permission set is owned by a profile, this field returns the ID of the Profile. If the permission set isn't owned by a profile, this field returns a null value. Available in API version 25.0 and later.</p>
UserLicenseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the UserLicense associated with this permission set. This field is nillable in API version 26.0 and later and available up to API version 37.0 In API version 38.0 and later, use <code>LicenseId</code>.</p>

Usage

Use the PermissionSet object to query existing permission sets.

For example, to search for all permission sets that contain the "Modify All Data" permission:

```
SELECT Name, PermissionsModifyAllData
FROM PermissionSet
WHERE PermissionsModifyAllData=true
```

When combined with the PermissionSetAssignment object, you can create a nested query that returns all users assigned to a particular permission like "Modify All Data":

```
SELECT Name, (SELECT AssigneeId FROM Assignments)
FROM PermissionSet
WHERE PermissionsModifyAllData=true
```

You can also create a permission set, or delete a permission set if it isn't assigned to a user.

User Licenses

The user license controls the permissions that are available in a permission set.

Every permission set can be associated with a user license or permission set license. If you plan to assign a permission set to multiple users with different user and permission set licenses, leave `LicenseId` empty. If only users with one type of license will use this permission set, set the `LicenseId` to that single user or permission set license. If you want a permission set associated with a permission set license, then set `LicenseId` to the permission set license. To get the `LicenseId`, run this query:

```
SELECT Id, Name  
FROM UserLicense
```

Alternatively, to query a user or profile for the `LicenseId`.

```
SELECT Id, Profile.UserLicenseId  
FROM User
```

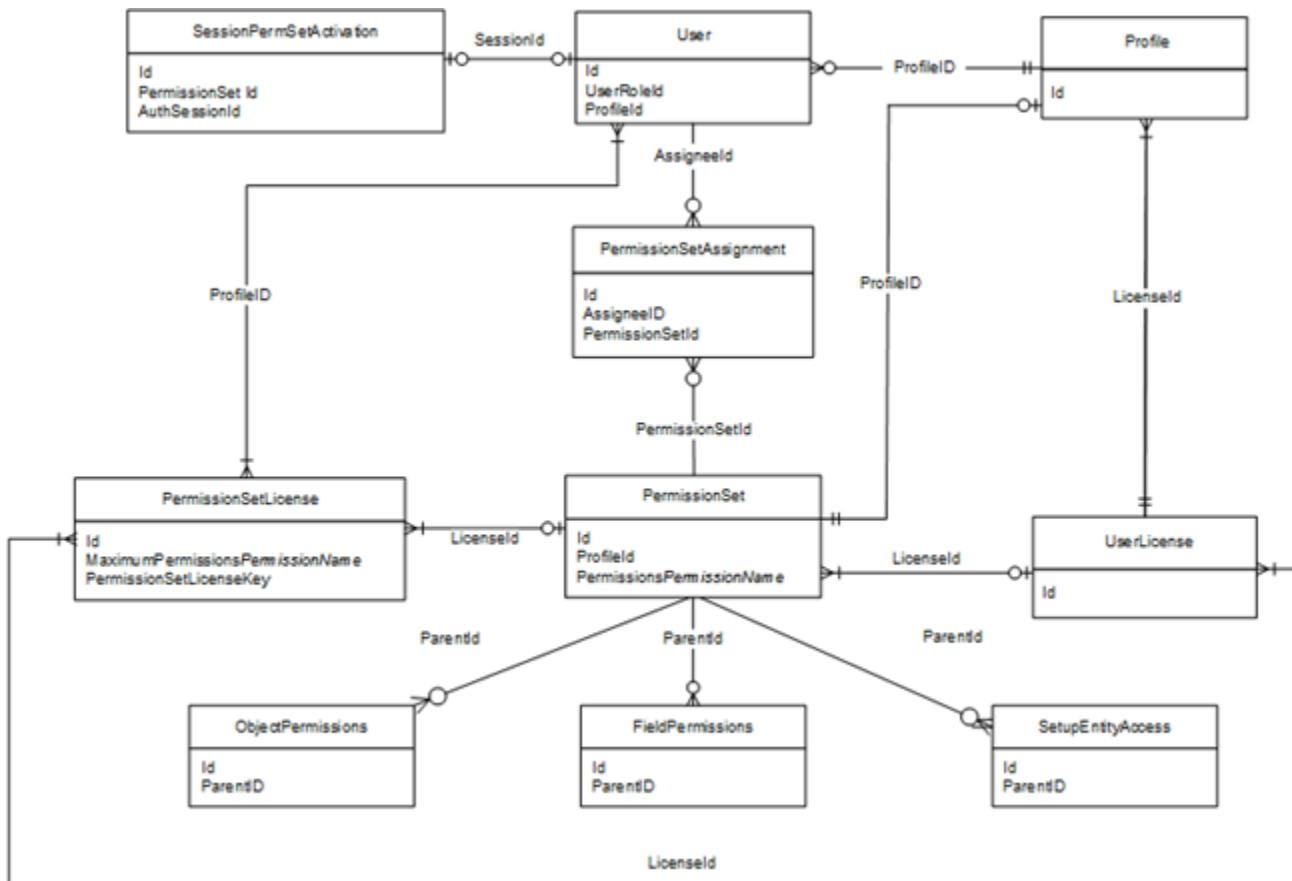
Child Objects

When using the API, think of each permission set or related set of access controls as an empty container that you fill with permission records.

In the API, a permission set can contain user, object, and field permissions, as well as setup entity access settings for other settings, such as Apex classes.

- `ObjectPermissions` and `FieldPermissions` objects are available in API version 24.0 and later.
- The `SetupEntityAccess` object is available in API version 25.0 and later.

Only user permissions are managed in the `PermissionSet` API object; all other permission types are managed in child API objects.



In these child objects, access is stored in a record, while the absence of a record indicates no access. To return a record in a SOQL query, a minimum permission or setting is required for each child object.

Because permissions are stored in related objects, it's important to understand what questions to ask when using SOQL. For example, you might want to know which permission sets have "Delete" on an object or have the right to approve a return merchandise authorization (where the approval checkbox is controlled with field permissions). Asking the right questions when using SOQL with permission sets ensures that you get the information you need to make an informed decision, such as whether to migrate permissions or assign a permission set to a user.

For example, the following returns all permission sets where the "Read" permission is enabled for the Merchandise__c object.

```
SELECT SObjectType, ParentId, PermissionsRead
FROM ObjectPermissions
WHERE PermissionsRead = True AND SObjectType = 'Merchandise__c'
```

You can query for all permission sets that have "Read" on an object. However, you can't query for permission sets that have no access on an object, because no records exist for that object. For example, the following returns no records because the object must have at least "Read" to return any records.

```
SELECT SObjectType, ParentId, PermissionsRead
FROM ObjectPermissions
WHERE PermissionsRead = False AND SObjectType = 'Merchandise__c'
```

If you have at least the “Read” permission on an object, you can create a conditional query on other permissions in the same object. For example, the following returns any records where the object has at least the “Read” permission but not the “Edit” permission.

```
SELECT ParentId, PermissionsRead, PermissionsEdit
FROM ObjectPermissions
WHERE PermissionsEdit = False AND SObjectType = 'Merchandise__c'
```

To set an object or field permission to no access, delete the record that contains the permission. For example, to disable all object permissions in the Merchandise__c object for a particular permission set, first query to retrieve the ID of the object permission record.

```
SELECT Id
FROM ObjectPermissions
WHERE SObjectType = 'Merchandise__c'
```

Then delete the IDs returned from the query.

 **Note:** If you try to update the object or field permissions by setting all permissions to false, the permission record is automatically deleted; any subsequent queries for the record ID won’t return results and you must add a new permission record to grant access.

View a Permission Set with Nested Queries

You can build on the PermissionSet object using child relationships that show all of the permissions in a single permission set. For example, the following returns all permission sets and displays the “Transfer Leads” permission, as well as any “Read” permissions on any objects and fields.

```
SELECT Label, PermissionsTransferAnyLead,
(SELECT SObjectType, PermissionsRead FROM ObjectPerms),
(SELECT SObjectType, Field, PermissionsRead FROM FieldPerms)
FROM PermissionSet
```

Associated Profiles

In API version 25.0 and later, every profile is associated with a permission set that stores the profile’s user, object, and field permissions, as well as setup entity access settings. You can query permission sets that are owned by profiles but not modify them.

The following example returns all permission sets, including those owned by a profile.

```
SELECT Id, Label, ProfileId, Profile.Name
FROM PermissionSet
```

The following returns all permission sets except those owned by profiles.

```
SELECT Id, Label, ProfileId, Profile.Name, IsOwnedByProfile
FROM PermissionSet
WHERE IsOwnedByProfile = FALSE
```

Because permission sets have child objects in the API, you can query their values on permission sets owned by a profile. For example, the following returns all enabled object permission records for profiles only.

```
SELECT Id, ParentId, PermissionsRead, SObjectType, Parent.ProfileId
FROM ObjectPermissions
WHERE Parent.IsOwnedByProfile = TRUE
```

Once you have the IDs for permission sets that are owned by profiles as well as those not owned by profiles, you can use the [PermissionSetAssignment](#) object to determine whether users can access an object or field via a permission in their profile or any of their

permission sets. For example, the following SOQL query returns all users who have the “Read” permission on the Merchandise__c object and specifies whether the permission is granted through a profile or permission set.

```
SELECT Assignee.Name, PermissionSet.Id, PermissionSet.isOwnedByProfile
FROM PermissionSetAssignment
WHERE PermissionSetId
IN (SELECT ParentId
FROM ObjectPermissions
WHERE SObjectType = 'Merchandise__c' AND PermissionsRead = true)
```

 **Note:** For permission sets that are owned by profiles, don't use any Name and Label values that are returned in a query, as they can change at any time.

SEE ALSO:

[ObjectPermissions](#)
[FieldPermissions](#)
[SetupEntityAccess](#)
[PermissionSetAssignment](#)
[Profile](#)

PermissionSetAssignment

Represents the association between a User and a PermissionSet. This object is available in API version 22.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
AssigneeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User to assign the permission set specified in <code>PermissionSetId</code>.</p>
PermissionSetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the PermissionSet to assign to the user specified in <code>AssigneeId</code>.</p>

Usage

Finding Permission Set Assignments

Use the `PermissionSetAssignment` object to query permission set assignments to find out which permission sets are assigned to which users. Because each user may be assigned to many permission sets and each permission set may be assigned to many users, each `PermissionSetAssignment` ID represents the association of a single user and single permission set.

For example, to search for all of the permission sets assigned to a particular user:

```
SELECT Id, PermissionSetId  
FROM PermissionSetAssignment  
WHERE AssigneeId = '005600000017cKt'
```

To search for all users assigned to a particular permission set:

```
SELECT Id, AssigneeId  
FROM PermissionSetAssignment  
WHERE PermissionSetId = '0PS3000000000e'
```

You can also create a new permission set assignment, or use `delete` to remove a permission set that's assigned to a user. To update an assignment, delete an existing assignment and insert a new one.

User Licenses

When assigning a permission set, if the `PermissionSet` has a `UserLicenseId`, its `UserLicenseId` and the Profile `UserLicenseId` must match. To determine a user's license assignment, query the user's profile and then query the profile's license.

For example, to find a user's profile ID:

```
SELECT Id, ProfileId  
FROM User  
WHERE Id = '005D0000001GMAT'
```

To find a permission set's `UserLicenseId`:

```
SELECT Id, LicenseId  
FROM PermissionSet  
WHERE Id = '0PS3000000000e'
```

If the IDs match, the assignment succeeds.

To find all the permission sets with no license that are assigned to any user:

```
SELECT Id, Assignee.Name, PermissionSet.Name  
FROM PermissionSetAssignment  
WHERE PermissionSet.LicenseId = null
```

SEE ALSO:

[PermissionSet](#)

PermissionSetLicense

Represents a license that's used to enable one or more users to receive a specified permission without changing their profile or reassigning profiles. You can use permission set licenses to grant access, but not to deny access. This object is available in API version 29.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
ExpirationDate	<p>Type date</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The date at which the permission set license expires.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of the permission set license.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The label of the permission set license. Label is Permission Set License Label.</p>

Field Name	Details
MaximumPermissionsPermissionName	Type boolean
	Properties Filter,
	Description One field for each permission. For example, <i>MaximumPermissionsIdentityConnect</i> corresponds to the “Use Identity Connect” permission. If <code>true</code> , this PermissionSetLicense grants the specified permission. The number of fields varies depending on the permissions available for the organization.
MaximumPermissionsShowCompanyNameAsUserBadge	Type boolean
	Properties Filter
	Description When on, a user’s company name, if available, will be displayed in place of the community role.
PermissionSetLicenseKey	Type string
	Properties Filter, Group, idLookup, Sort
	Description A string that uniquely identifies a particular permission set license.
Status	Type picklist
	Properties Filter, Group, Restricted picklist, Sort
	Description The status of a permission set license. If <code>Active</code> , the permission set license is available. If <code>Disabled</code> , the permission set license has expired.
TotalLicenses	Type int
	Properties Filter, Group, Sort
	Description The total number of this permission set license that are available to your organization.

Field Name	Details
UsedLicenses	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of this permission set license that are currently assigned to users.</p>

Usage

Users with the “View Setup and Configuration” permission can use the PermissionSetLicense object to view the set of currently defined permission set licenses in your organization.

Use the PermissionSetLicense object to query existing permission licenses.

For example, to return a list of all active permission set licenses:

```
SELECT MasterLabel
FROM PermissionSetLicense
WHERE Status = 'Active'
```

When combined with the PermissionSetLicenseAssign object, you can create a nested query that returns all users assigned to a particular permission set license like “Identity Connect”:

```
SELECT MasterLabel, (SELECT AssigneeId FROM PermissionSetLicenseAssignments)
FROM PermissionSetLicense
WHERE MaximumPermissionsIdentityConnect=true
```

SEE ALSO:

[PermissionSetLicenseAssign](#)

PermissionSetLicenseAssign

Represents the association between a User and a PermissionSetLicense. This object is available in API version 29.0 and later.

 **Note:** The relationship name for PermissionSetLicenseAssign is *PermissionSetLicenseAssignments*.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
AssigneeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User to assign the permission set license specified in <code>PermissionSetLicenseId</code>.</p>
PermissionSetLicenseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the permission set license the user is assigned to.</p>

Usage

Use the `PermissionSetLicenseAssign` object for querying permission set license assignments to find out which permission set licenses are assigned to which users. Because each user can be assigned to many permission set licenses, each `PermissionSetLicenseAssign` ID represents the association of a single user and single permission set license.

For example, to search for all of the permission sets assigned to a particular user:

```
SELECT Id, PermissionSetLicenseId
FROM PermissionSetLicenseAssign
WHERE AssigneeId = '005D0000001RFek'
```

To search for all users assigned to a particular permission set license:

```
SELECT AssigneeId
FROM PermissionSetLicenseAssign
WHERE PermissionSetLicenseId = '0PID00000003mwOAA'
```

You can also create a new permission set license assignment, or use delete to remove a permission set license that's been assigned to a user. To update an assignment, delete an existing assignment and insert a new one.

SEE ALSO:

[PermissionSetLicense](#)

PlatformAction

`PlatformAction` is a virtual read-only object. It enables you to query for actions displayed in the UI, given a user, a context, device format, and a record ID. Examples include standard and custom buttons, quick actions, and productivity actions.

Supported Calls

`query()`

Fields

Field	Details
ActionListContext	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Required. The list context this action applies to. Valid values are:</p> <ul style="list-style-type: none"> • Assistant • BannerPhoto • Chatter • Dockable • FeedElement • FlexiPage • Global • ListView • ListViewDefinition • ListViewRecord • Lookup • MruList • MruRow • ObjectHomeChart • Photo • Record • RecordEdit • RelatedList • RelatedListRecord
ActionTarget	<p>Type string</p> <p>Properties Nillable</p> <p>Description The URL to invoke or describe the action when the action is invoked. If the action is a standard button overridden by a Visualforce page, the ActionTarget returns the URL of the Visualforce page, such as <code>/apex/pagename</code>.</p>

Field	Details
	<p>This field is available in API version 35.0 and later.</p>
Action targetType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of the target when this action is triggered. Valid values are:</p> <ul style="list-style-type: none">• Describe—applies to actions with a user interface, such as quick actions• Invoke—applies to actions with no user interface, such as action links or invocable actions• Visualforce—applies to standard buttons overridden by a Visualforce page
Action target url	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL to invoke or describe the action when the action is invoked. This field is deprecated in API version 35.0 and later. Use <code>Action target</code> instead.</p>
Category	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Applies only to action links. Denotes whether the action link shows up in the feed item list of actions or the overflow list of actions. Valid values are:</p> <ul style="list-style-type: none">• Primary• Overflow
Confirmation message	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Applies only to action links. The message to display before the action is invoked. Field is null if no confirmation is required before invoking the action.</p>
Device format	<p>Type string</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies which action icon the PlatformAction query returns. If this field isn't specified, it defaults to Phone. Valid values are:</p> <ul style="list-style-type: none">• Aloha• Desktop• Phone• Tablet
ExternalId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The unique ID for the PlatformAction. If the action doesn't have an ID, its API name is used.</p>
GroupId	<p>Type ID</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The unique ID of a group of action links.</p>
IconContentType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The content type—such as .jpg, .gif, or .png—of the icon for this action. Applies to both custom and standard icons assigned to actions.</p>
IconHeight	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The height of the icon for this action. Applies only to standard icons.</p>
IconUrl	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL of the icon for this action.</p>
IconWidth	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The width of the icon for this action. Applies only to standard icons.</p>
InvocationStatus	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The status of the action within the feed item. Applies to action links only. Valid values are:</p> <ul style="list-style-type: none"> • Failed • New • Pending • Successful
InvokedByUserId	<p>Type ID</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who most recently invoked this action within the current feed item. Applies to action links only.</p>
IsGroupDefault	<p>Type boolean</p> <p>Properties Filter, Group, Sort</p> <p>Description Denotes whether this action is the default in an action link group. False for other action types. Applies to action links only.</p>
IsMassAction	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the action can be performed on multiple records. This field is available in API version 38.0 and later.</p>
Label	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The label to display for this action.</p>
PrimaryColor	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The primary color of the icon for this action.</p>
RelatedListRecordId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the ID of a record in an object's related list. This field is available in API version 38.0 and later.</p>
RelatedSourceEntity	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When the <code>ActionListContext</code> is <code>RelatedList</code> or <code>RelatedListRecord</code>, this field represents the API name of the related list to which the action belongs.</p>
Section	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description The section of the user interface the action resides in. Applicable only to Lightning Experience. Valid values are:</p> <ul style="list-style-type: none"> • ActivityComposer • CollaborateComposer • NotesComposer • Page • SingleActionLinks <p>This field is available in API version 35.0 and later.</p>
SourceEntity	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. The object or record with which this action is associated.</p>
Subtype	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The subtype of the action. For quick actions, the subtype is <code>QuickActionType</code>. For custom buttons, the subtype is <code>WebLinkTypeEnum</code>. For action links, subtypes are <code>Api</code>, <code>ApiAsync</code>, <code>Download</code>, and <code>Ui</code>. Standard buttons and productivity actions have no subtype.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of the action. Valid values are:</p> <ul style="list-style-type: none"> • <code>ActionLink</code>—An indicator on a feed element that targets an API, a web page, or a file, represented by a button in the Salesforce Chatter feed UI. • <code>CustomButton</code>—When clicked, opens a URL or a Visualforce page in a window or executes JavaScript. • <code>InvocableAction</code> • <code>ProductivityAction</code>—Productivity actions are predefined by Salesforce and are attached to a limited set of objects. You can't edit or delete productivity actions. • <code>QuickAction</code>—A global or object-specific action.

Field	Details
	<ul style="list-style-type: none"> • StandardButton—A predefined Salesforce button such as New, Edit, and Delete.

Usage

PlatformAction can be described using `describeSObject()`.

You can directly query for PlatformAction. For example, this query returns all fields for actions associated with each of the records of the listed objects:

```
SELECT ExternalId, ActionTargetType, ActionTargetUrl, ApiName, Category,
       ConfirmationMessage, ExternalId, GroupId, UiTheme, IconUrl, IconContentType,
       IconHeight, IconWidth, PrimaryColor, InvocationStatus, InvokedById,
       IsGroupDefault, Label, LastModifiedDate, Subtype, SourceEntity, Type
FROM PlatformAction
WHERE SourceEntity IN ('001xx000003DGsH', '001xx000003DHBq', 'Task', 'Global') AND
      ActionListContext = 'Record';
```

 **Note:** To query PlatformAction, provide the `ActionListContext` and `SourceEntity`. If you query for `ActionListContext` with a value of `RelatedList`, and don't specify a `RelatedSourceEntity`, the query returns the API name of the related list.

This query uses multiple `ActionListContext` values in its `WHERE` clause to return all actions in the Lightning Experience user interface (`DeviceFormat = 'Desktop'`) for the specified object:

```
SELECT ActionListContext, Label, Type, Subtype, Section, SourceEntity,
       RelatedSourceEntity, ActionTarget, ActionTargetType, ApiName, Category,
       ConfirmationMessage, DeviceFormat, ExternalId, GroupId, IconContentType,
       IconHeight, IconUrl, IconWidth, Id, InvocationStatus, InvokedById,
       IsGroupDefault, LastModifiedDate, PrimaryColor
FROM PlatformAction
WHERE ActionListContext IN ('Record', 'Chatter', 'RelatedList') AND
      SourceEntity = '001xx000003DlvX' AND
      DeviceFormat = 'Desktop'
```

PresenceUserConfig

Represents a configuration that determines a presence user's settings. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
Capacity	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The maximum number of work assignments that can be pushed to an agent at a time.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the presence configuration.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the presence configuration.</p>
OptionsIsAutoAcceptEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
	<p>Description Indicates whether work items that are routed to agents are automatically accepted (<code>true</code>) or not (<code>false</code>). Available only if <code>OptionsIsDeclineEnabled</code> is set to <code>false</code>.</p>
OptionsIsDeclineEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether agents can decline work items that are routed to them (<code>true</code>) or not (<code>false</code>). Available only if <code>OptionsIsAutoAcceptEnabled</code> is set to <code>false</code>.</p>
OptionsIsDeclineReasonEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether agents can select a reason for declining work requests (<code>true</code>) or not (<code>false</code>). This can be selected only if decline reasons are enabled.</p>
OptionsIsDisconnectSoundEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether a sound is played when agents are disconnected from Omni-Channel (<code>true</code>) or not (<code>false</code>).</p>
OptionsIsRequestSoundEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether a sound plays with incoming work requests (<code>true</code>) or not (<code>false</code>). Set to <code>true</code> by default.</p>
PresenceStatusOnDeclineId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p>

Field	Details
PresenceStatusOnPushTimeoutId	<p>Description The ID of the presence status that's automatically assigned to the agent when the agent declines a work item. Available only if <code>OptionsIsDeclineEnabled</code> is set to <code>true</code>.</p>
Type	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
PresenceStatusOnPushTimeoutId	<p>Description The ID of the presence status that's automatically assigned to the agent when the agent doesn't respond to a work item before push timeout occurs. Available in API version 36.0 and later.</p>

PresenceUserConfigProfile

Represents a configuration that determines the settings that are assigned to presence users who are assigned to a specific profile. User-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
PresenceUserConfigId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description If an individual user is also assigned a presence configuration through the <code>PresenceUserConfigProfile</code>, this configuration will override that.</p>
ProfileId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the profile that's associated with this presence configuration. A profile can be associated with only one presence configuration.</p>

PresenceUserConfigUser

Represents a configuration that determines the settings that are assigned to a presence user. These user-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
PresenceUserConfigId	Type reference Properties Create, Filter, Group, Sort Description The ID of the presence configuration.
UserId	Type reference Properties Create, Filter, Group, Sort Description The ID of the user who's associated with this presence configuration. A user can be associated with only one presence configuration.

Pricebook2

Represents a price book that contains the list of products that your org sells.



Note: Price books are represented by Pricebook2 objects. As of API version 8.0, the Pricebook object is no longer available. Requests containing Pricebook are refused, and responses do not contain the Pricebook object.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of this object.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this object is active (<code>true</code>) or not (<code>false</code>). Inactive objects are hidden in many areas in the user interface. You can change this field's value as often as necessary. Label is Active.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
IsStandard	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this object is the standard price book for the org (<code>true</code>) or not (<code>false</code>). Every org has one standard price book—all other price books are custom price books.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of this object. This field is read-only for the standard price book. Label is Price Book Name.</p>

Usage

A price book is a list of products that your org sells.

- Each org has one standard price book that defines the standard or generic list price for each product or service that it sells.
- An org can have multiple custom price books to use for specialized purposes, such as for discounts, different channels or markets, or select accounts or opportunities. While your client application can create, delete, and update custom price books, your client application can update only the standard price book.
- For some orgs, the standard price book is the only price needed. If you need to set up other price books, you can reference the standard price book when setting up list prices in custom price books.

Use this object to query standard and custom price books that have been configured for your org. A common use of this object is to allow your client application to obtain valid Pricebook2 object IDs for use when configuring PricebookEntry records via the API.

Your client application can perform the following tasks on PricebookEntry objects:

- Query
- Create for the standard pricebook or custom pricebooks.
- Update
- Delete
- Change the `IsActive` field when creating or updating records

PriceBook2, Product2, and PricebookEntry Relationships

In the API:

- Price books are represented by Pricebook2 objects (as of version 8.0, the Pricebook object is no longer available).
- Products are represented by Product2 objects (as of version 8.0, the Product object is no longer available).
- Each price book contains zero or more entries (represented by PricebookEntry records) that specify the products that are associated with the price book. A price book entry defines the price for which you sell a product at a particular currency.

These objects are defined only for those orgs that have products enabled as a feature. If the org doesn't have the products feature enabled, the Pricebook2 object doesn't appear in the `describeGlobal()` call, and you can't access it via the API.

If you delete a Pricebook2 while a line item references PricebookEntry in the price book, the line item is unaffected, but the Pricebook2 is archived and unavailable from the API.

For a visual diagram of the relationships between Pricebook2 and other objects, see Product and Schedule Objects.

Price Book Setup

The process of setting up a price book via the API usually means:

1. Initially loading product data from your org into Product2 objects (creating a Product2 record for each product that you want to add).
2. For each Product2 object, creating a PricebookEntry that links the Product2 object to the standard Pricebook2. You need to define a standard price for a product at a given currency (if you have multicurrency enabled), before defining a price for that product in the same currency in a custom price book.
3. Creating a custom Pricebook2.
4. Querying the Pricebook2 object to obtain their IDs.

5. For each Pricebook2 object, creating a PricebookEntry for every Product2 that you want to add, specifying unique properties for each PricebookEntry (such as the UnitPrice and CurrencyIsoCode) as needed.

Code Sample—Java

```
public void pricebookSample() {  
    try {  
        //Create a custom pricebook  
        Pricebook2 pb = new Pricebook2();  
        pb.setName("Custom Pricebook");  
        pb.setIsActive(true);  
        SaveResult[] saveResults = connection.create(new SObject[]{pb});  
        pb.setId(saveResults[0].getId());  
  
        // Create a new product  
        Product2 product = new Product2();  
        product.setIsActive(true);  
        product.setName("Product");  
        saveResults = connection.create(new SObject[]{product});  
        product.setId(saveResults[0].getId());  
  
        // Add product to standard pricebook  
        QueryResult result = connection.query(  
            "select Id from Pricebook2 where isStandard=true"  
        );  
        SObject[] records = result.getRecords();  
        String stdPbId = records[0].getId();  
  
        // Create a pricebook entry for standard pricebook  
        PricebookEntry pbe = new PricebookEntry();  
        pbe.setPricebook2Id(stdPbId);  
        pbe.setProduct2Id(product.getId());  
        pbe.setIsActive(true);  
        pbe.setUnitPrice(100.0);  
        saveResults = connection.create(new SObject[]{pbe});  
  
        // Create a pricebook entry for custom pricebook  
        pbe = new PricebookEntry();  
        pbe.setPricebook2Id(pb.getId());  
        pbe.setProduct2Id(product.getId());  
        pbe.setIsActive(true);  
        pbe.setUnitPrice(100.0);  
        saveResults = connection.create(new SObject[]{pbe});  
    } catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
}
```

SEE ALSO:

[Product and Schedule Objects](#)

Pricebook2History

Represents historical information about changes that have been made to the standard fields of the associated Pricebook2, or to any custom fields with history tracking enabled. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is always read-only.

Fields

Field	Details
Pricebook2Id	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Pricebook2 associated with this record.</p>
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Name of the price book field that was modified, or a special value to indicate some other modification to the price book.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). This is a standard system field. Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field	Details
	<p>Description New value of the modified price book field. Maximum of 255 characters.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description Previous value of the modified price book field. Maximum of 255 characters.</p>

Usage

Price book history entries are indirectly created each time a price book is modified.

Two rows are added to this record when foreign key fields change. One row contains the foreign key object names that display in the online application. For example, Jane Doe is recorded as the name of a Contact. The other row contains the actual foreign key ID that is only returned to and visible from the API.

This object respects field level security on the parent object.

SEE ALSO:

[Pricebook2](#)

PricebookEntry

Represents a product entry (an association between a Pricebook2 and Product2) in a price book.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), update(), upsert()
```

Fields

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p>

Field	Details
	<p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this price book entry is active (<code>true</code>) or not (<code>false</code>). Although you can never delete PricebookEntry records, your client application can set this flag to <code>false</code>. Inactive PricebookEntry records are hidden in many areas in the user interface. You can change this flag on a PricebookEntry record as often as necessary.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether this PricebookEntry record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of this PricebookEntry record. This read-only field references the value in the Name field of the Product2 record. Label is Product Name.</p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Pricebook2 record with which this record is associated. This field must be specified when creating Pricebook2 records. It can't be changed in an update.</p>
Product2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field	Details
	Description
	Required. ID of the Product2 record with which this record is associated. This field must be specified when creating Product2 records. It can't be changed in an update.
ProductCode	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Product code for this record. This read-only field references the value in the ProductCode field of the associated Product2 record.
UnitPrice	Type currency
	Properties
	Create, Filter, Sort, Update
	Description
	Required. Unit price for this price book entry. You can specify a value only if UseStandardPrice is set to false . Label is List Price .
UseStandardPrice	Type boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates whether this price book entry uses the standard price defined in the standard Pricebook2 record (true) or not (false). If set to true , then the UnitPrice field is read-only, and the value will be the same as the UnitPrice value in the corresponding PricebookEntry in the standard price book (that is, the PricebookEntry record whose Pricebook2Id refers to the standard price book and whose Product2Id and CurrencyIsoCode are the same as this record). For PricebookEntry records associated with the standard Pricebook2 record, this field must be set to true .

Usage

Use this object to define the association between your organization's products (Product2) and your organization's standard price book or to other, custom-defined price books (Pricebook2). Create one PricebookEntry record for each standard or custom price and currency combination for a product in a Pricebook2.

When creating these records, you must specify the IDs of the associated Pricebook2 record and Product2 record. Once created, your client application can't update these IDs.

This object is defined only for those organizations that have products enabled as a feature. If the organization does not have the products feature enabled, then the PricebookEntry object does not appear in the `describeGlobal()` call, and you can't access it.

If you delete a PricebookEntry while a line item references it, the line item is unaffected, but the PricebookEntry will be archived and unavailable from the API.

You must load the standard price for a product before you are permitted to load its custom price(s).

SEE ALSO:

[Object Basics](#)

ProcessDefinition

Represents the definition of a single approval process.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `search()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description A description of this process, with a maximum of 3,000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The external name of the process; the name seen by users.</p>
LockType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field	Details
	<p>Description</p> <p>The type of lock applied to the record being approved. When a record is in the approval process, it's always locked, and only an administrator can edit it. However, the currently assigned approver can also be allowed to edit the record.</p> <ul style="list-style-type: none"> • Total • Admin • Owner • Workitem • Node • none
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The unique process name, used internally.</p>
State	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The current state of this process.</p> <ul style="list-style-type: none"> • Active • Inactive • Obsolete
TableEnumOrId	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>Specifies the object associated with the approval process, such as Account or Contact.</p>
Type	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>The type of this process.</p>

Field	Details
	<ul style="list-style-type: none"> Approval Process—Used to control the action taken for a record. State-based Process—Used internally to track various control processes, such as for developing Salesforce Knowledge articles.

Usage

Use this object to read the description of an approval process. The definition is read-only.

ProcessInstance

Represents an instance of a single, end-to-end approval process. Use this and the node, step, and workitem process instance objects to create approval history reports.

 **Note:** Exceptions apply to approval history data retrieved with this object and available only via the SOAP API. For each approval process instance that was pending when Summer '14 became available for your organization, some field values are never populated or are populated only after the approval process instance is next acted upon—such as when a user approves, rejects, or reassigns an approval request—after the Summer '14 rollout.

For approval process instances that were completed before the Summer '14 rollout, all ProcessInstance fields are automatically populated, with one exception: `CompletedDate` is never populated for approval process instances that were completed before January 1, 2013. For approval process instances that were pending during the Summer '14 rollout, all ProcessInstance fields are automatically populated, with two exceptions: `CompletedDate` and `LastActorId` are populated only after the approval process instance is complete.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
CompletedDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The completion date and time of the approval process. The <code>Elapsed Time Day</code>, <code>Elapsed Time Hours</code>, and <code>Elapsed Time Minutes</code> field values are calculated using <code>CompletedDate</code>.</p>

Field	Details
Elapsed Time In Days	<p>Type double</p> <p>Properties Filter, Sort</p> <p>Description The total elapsed time in days between when the approval process instance was started and now.</p>
Elapsed Time In Hours	<p>Type double</p> <p>Properties Filter, Sort</p> <p>Description The total elapsed time in hours between when the approval process instance was started and now.</p>
Elapsed Time In Minutes	<p>Type double</p> <p>Properties Filter, Sort</p> <p>Description The total elapsed time in minutes between when the approval process instance was started and now.</p>
Last Actor Id	<p>Type reference</p> <p>Properties Group, Filter, Sort</p> <p>Description The last actor that approved, rejected, or recalled the process.</p>
Process Definition Id	<p>Type reference</p> <p>Properties Defaulted on create, Group, Filter, Sort</p> <p>Description The ID of this approval process instance.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field	Details
	<p>Description The status of this approval process instance, for example Started, Pending, or Approved.</p>
TargetObjectId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the object affected by this approval process instance.</p>

Usage

Use this object to query or retrieve an approval process.

The following SOQL query returns details for all the ProcessInstanceStep records related to individual ProcessInstance records. The nested query references `Steps`, which is the child `relationshipName` for ProcessInstanceStep in the ProcessInstance object.

```
SELECT Id, (SELECT Id, StepStatus, Comments FROM Steps)
FROM ProcessInstance
```

The following SOQL query returns details for all the ProcessInstanceWorkitem records related to individual ProcessInstance records. The nested query references `Workitems`, which is the child `relationshipName` for ProcessInstanceWorkitem in the ProcessInstance object.

```
SELECT Id, (SELECT Id, ActorId, ProcessInstanceId FROM Workitems)
FROM ProcessInstance
```

ProcessInstanceHistory can help provide a unified read-only view of the ProcessInstanceStep and ProcessInstanceWorkitem objects.

SEE ALSO:

[ProcessInstanceHistory](#)

[ProcessInstanceStep](#)

[ProcessInstanceWorkitem](#)

ProcessInstanceHistory

This read-only object shows all steps and pending approval requests associated with an approval process (ProcessInstance).

Supported Calls

`describeSObjects()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
ActorId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who is currently assigned to this ProcessInstance.</p>
Comments	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Comments for a ProcessInstanceStep. This field doesn't apply to ProcessInstanceWorkitem records.</p>
Elapsed Time In Days	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in days between when the approval process instance was started and when it was completed.</p>
Elapsed Time In Hours	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in hours between when the approval process instance was started and when it was completed.</p>
Elapsed Time In Minutes	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in minutes between when the approval process instance was started and when it was completed.</p>

Field	Details
IsPending	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the ProcessInstance is pending (<code>true</code>) or not (<code>false</code>).</p>
OriginalActorId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who was originally assigned this ProcessInstance.</p>
ProcessInstanceId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the ProcessInstance.</p>
ProcessNodeId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of this step.</p>
RemindersSent	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number of reminders that have been sent. Default is 0 (zero).</p>
StepStatus	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates the current status of the ProcessInstanceStep.</p>

Field	Details
TargetObjectId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the object being approved.</p>

Usage

This object helps you replicate the related list functionality of the Salesforce user interface for approval processes. Use ProcessInstanceHistory for a unified read-only view of the ProcessInstanceStep and ProcessInstanceWorkitem objects. You can't query ProcessInstanceHistory. Instead, you can query ProcessInstanceHistory by including it in a nested query on the parent ProcessInstance object. For example, the following SOQL query returns all the ProcessInstanceHistory records related to individual ProcessInstance records. The nested query references StepsAndWorkitems, which is the child relationshipName for ProcessInstanceHistory in the ProcessInstance object.

```
SELECT Id, (SELECT Id, StepStatus, Comments FROM StepsAndWorkitems)
FROM ProcessInstance
```

This object respects field-level security on the parent object.

SEE ALSO:

[ProcessInstance](#)
[ProcessInstanceStep](#)
[ProcessInstanceWorkitem](#)

ProcessInstanceStep

Represents one work item in an approval process (ProcessInstance).

 **Note:** Exceptions apply to approval history data retrieved with this object and available only via the SOAP API. For each approval process instance that was pending when Summer '14 became available for your organization, some field values are never populated or are populated only after the approval process instance is next acted upon—such as when a user approves, rejects, or reassigns an approval request—after the Summer '14 rollout.

ProcessInstanceStep fields are never populated for approval process instances that were completed before the Summer '14 rollout. For approval process instances that were pending during the Summer '14 rollout, all ProcessInstanceStep fields are populated only after the approval process instance is next acted upon after the Summer '14 rollout.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
ActorId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who is currently assigned to this approval step.</p>
Comments	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description Limit: 4,000 bytes.</p>
Elapsed Time In Days	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in days since this step was started.</p>
Elapsed Time In Hours	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in hours since this step was started.</p>
Elapsed Time In Minutes	<p>Type double</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The total time in minutes since this step was started.</p>

Field	Details
OriginalActorId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user who was originally assigned to this approval step.</p>
ProcessInstanceId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the ProcessInstance that this approval step belongs to.</p>
StepNodeId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the node currently assigned to this approval step.</p>
StepStatus	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The current status of this approval step.</p> <ul style="list-style-type: none"> • Approved • Fault • Held • NoResponse • Pending • Reassigned • Rejected • Removed • Started <p>If the approval step requires unanimous approval and one approver rejects the request, the value of this field for the other approvers changes to NoResponse. Likewise, if approval is based on the first response and an approver responds, the value of this field for the other approvers changes to NoResponse.</p>

Usage

Query or retrieve a new step in an approval process (ProcessInstance).

SEE ALSO:

[ProcessInstance](#)
[ProcessInstanceHistory](#)
[ProcessInstanceWorkitem](#)

ProcessInstanceNode

Represents a step in an instance of an approval process. Compare to ProcessNode, which describes the step in a process definition. Use this object to retrieve approval history.

 **Note:** Exceptions apply to approval history data retrieved with this object and available only via the SOAP API. For each approval process instance that was pending when Summer '14 became available for your organization, some field values are never populated or are populated only after the approval process instance is next acted upon—such as when a user approves, rejects, or reassigns an approval request—after the Summer '14 rollout.

ProcessInstanceNode fields are never populated for approval process instances that were completed before the Summer '14 rollout. For approval process instances that were pending during the Summer '14 rollout, all ProcessInstanceNode fields are populated only after the approval process instance is next acted upon after the Summer '14 rollout.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
CompletedDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The completion date and time of this step in the approval process. The <code>Elapsed Time Day</code>, <code>Elapsed Time Hours</code>, and <code>Elapsed Time Minutes</code> field values are calculated using <code>Completed Date</code>.</p>
ElapsedTimeInDays	<p>Type double</p>

Field	Details
	Properties
	Filter, Sort
	Description
	The total time in days since this step was started.
Elapsed Time In Hours	Type
	double
	Properties
	Filter, Sort
	Description
	The total time in hours since this step was started.
Elapsed Time In Minutes	Type
	double
	Properties
	Filter, Sort
	Description
	The total time in minutes since this step was started.
Last Actor Id	Type
	reference
	Properties
	Defaulted on create, Group, Filter, Sort
	Description
	The last actor that approved or rejected this step.
Node Status	Type
	picklist
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	The status of this approval instance, for example Started, Pending, or Approved.
Process Instance Id	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	The approval process this step is part of.
Process Node Id	Type
	reference

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The identifier for this step.</p>
ProcessNodeName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The name of this step.</p>

ProcessInstanceWorkitem

Represents a user's pending approval request.

 **Note:** Exceptions apply to approval history data retrieved with this object and available only via the SOAP API. For each approval process instance that was pending when Summer '14 became available for your organization, some field values are never populated or are populated only after the approval process instance is next acted upon—such as when a user approves, rejects, or reassigns an approval request—after the Summer '14 rollout.

ProcessInstanceWorkitem fields are never populated for approval process instances that were completed before the Summer '14 rollout. For approval process instances that were pending during the Summer '14 rollout, all ProcessInstanceWorkitem fields are populated after the approval process instance is next acted upon after the Summer '14 rollout, with three exceptions: `Elapsed Time In Days`, `Elapsed Time In Days`, and `Elapsed Time In Minutes` fields are never populated in ProcessInstanceWorkitem records for which equivalent ProcessInstanceStep records were created before the Summer '14 rollout.

For all other ProcessInstanceWorkitem records, these three fields are populated after the approval process instance is next acted upon after the Summer '14 rollout.

 **Note:** Because ProcessInstanceHistory combines fields from ProcessInstanceStep and ProcessInstanceWorkitem, you may notice incorrect elapsed times of 0 in ProcessInstanceHistory records because the elapsed time fields were never populated in the related ProcessInstanceWorkitem record.

Supported Calls

`delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
ActorId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the user who is currently responsible for approving an approval request.</p>
Elapsed Time In Days	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description The total time in days since this approval request was started.</p>
Elapsed Time In Hours	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description The total time in hours since this approval request was started.</p>
Elapsed Time In Minutes	<p>Type double</p> <p>Properties Filter, Nillable</p> <p>Description The total time in minutes since this approval request was started.</p>
Original Actor Id	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description ID of the user who was originally assigned this approval request.</p>
Process Instance Id	<p>Type reference</p> <p>Properties Filter, Group, Sort, Update</p>

Field	Details
	<p>Description ID of the ProcessInstance associated with this approval request.</p>

Usage

Use this object to manage a pending approval request for a user.

SEE ALSO:

[ProcessInstance](#)
[ProcessInstanceHistory](#)
[ProcessInstanceStep](#)

ProcessNode

Describes a step in a process definition. Compare to ProcessInstanceNode, which describes the step in a running process.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Portal and Communities users can't access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description A description of this node, no longer than 3,000 bytes.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p>

Field	Details
Description	
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique node name.</p>
ProcessDefinition	
ProcessDefinition	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the object affected by this approval instance.</p>

Usage

Use this object to get the description of a process definition.

The following SOQL query returns details for all the ProcessInstanceStep records related to individual ProcessInstance records. The nested query references `Steps`, which is the child `relationshipName` for ProcessInstanceStep in the ProcessInstance object.

```
SELECT Id, (SELECT Id, StepStatus, Comments FROM Steps)
FROM ProcessInstance
```

The following SOQL query returns details for all the ProcessInstanceWorkitem records related to individual ProcessInstance records. The nested query references `Workitems`, which is the child `relationshipName` for ProcessInstanceWorkitem in the ProcessInstance object.

```
SELECT Id, (SELECT Id, ActorId, ProcessInstanceId FROM Workitems)
FROM ProcessInstance
```

ProcessInstanceHistory can help provide a unified read-only view of the ProcessInstanceStep and ProcessInstanceWorkitem objects.

Product2

Represents a product that your org sells.

This object has several fields that are used only for quantity and revenue schedules (for example, annuities). Schedules are available only for orgs that have enabled the products and schedules features. If these features aren't enabled, the schedule fields don't appear in the DescribeSObjectResult, and you can't query, create, or update the fields.



Note: Products are represented by Product2 objects. As of API version 8.0, the Product object is no longer available. Requests containing Product are refused, and responses do not contain the Product object.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
CanUseQuantitySchedule	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the product can have a quantity schedule (<code>true</code>) or not (<code>false</code>). Label is Quantity Scheduling Enabled.</p>
CanUseRevenueSchedule	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the product can have a revenue schedule (<code>true</code>) or not (<code>false</code>). Label is Revenue Scheduling Enabled.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nullable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nullable</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the <code>ConnectionSentId</code> field is no longer supported. The <code>ConnectionSentId</code> field is still visible, but the value is null. You can use the new <code>PartnerNetworkRecordConnection</code> object to forward records to connections.</p>

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the org.</p>
DefaultPrice	<p>Type currency</p> <p>Properties Create, Filter, Update</p> <p>Description The default price for this record.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description A text description of this record. Label is Product Description.</p>
Family	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Name of the product family associated with this record. Product families are configured as picklists in the user interface. To obtain a list of valid values, call <code>describeSObjects()</code> and process the <code>DescribeSObjectResult</code> for the values associated with the <code>Family</code> field. Label is Product Family.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this record is active (<code>true</code>) or not (<code>false</code>). Inactive Product2 records are hidden in many areas in the user interface. You can change the <code>IsActive</code> flag on a Product2 object as often as necessary. Label is Active.</p>

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Default name of this record. Label is Product Name.</p>
NumberOfQuantityInstallments	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description If the product has a quantity schedule, the number of installments.</p>
NumberOfRevenueInstallments	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description If the product has a revenue schedule, the number of installments.</p>
ProductCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Default product code for this record. Your org defines the product code naming pattern.</p>
QuantityInstallmentPeriod	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description If the product has a quantity schedule, the amount of time covered by the schedule.</p>
QuantityScheduleType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of the quantity schedule, if the product has one.</p>
RecalculateTotalPrice	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Changes behavior of OpportunityLineItem calculations when a line item has child schedule rows for the <code>Quantity</code> value. When enabled, if the rollup quantity changes, then the quantity rollup value is multiplied against the sales price to change the total price.</p>
RevenueInstallmentPeriod	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description If the product has a revenue schedule, the period of time covered by the schedule.</p>
RevenueScheduleType	<p>Type picklist</p>

Field	Details
Properties	Filter, Group, Nillable, Restricted picklist, Sort
Description	The type of the revenue schedule, if the product has one.

Schedule Enabled Flags

When enabling the schedules feature, orgs can decide whether to enable quantity schedules, revenue schedules, or both. In addition, you can use the API to control quantity and revenue scheduling at the product level via the `CanUseQuantitySchedule` and `CanUseRevenueSchedule` flags. A value of `true` for either flag indicates that the product and any `OpportunityLineItems` can have a schedule of that type. These flags can be set when creating or updating `Product2` records.

Default Schedule Fields

The remaining schedule fields for this object define default schedules. Default schedule values are used to create an `OpportunityLineItemSchedule` when an `OpportunityLineItem` is created for the Product.

The default schedule fields support the following valid values (all fields are also nillable).

Field	Valid Values
<code>RevenueScheduleType</code>	Divide, Repeat
<code>RevenueInstallmentPeriod</code>	Daily, Weekly, Monthly, Quarterly, Yearly
<code>NumberOfRevenueInstallments</code>	Integer between 1 to 150, inclusive.
<code>QuantityScheduleType</code>	Divide, Repeat
<code>QuantityInstallmentPeriod</code>	Daily, Weekly, Monthly, Quarterly, Yearly
<code>NumberOfQuantityInstallments</code>	Integer between 1 to 150, inclusive

When you attempt to set the schedule fields when creating or updating, the API applies cross-field integrity checks. The integrity requirements are:

- If the schedule type is nil, the installment period and number of installments must be nil.
- If the schedule type is set to any value, then the installment period and number of installments must be non-nil.

Any create or update that fails these integrity checks is rejected with an error.

These default schedule fields, as well as `CanUseQuantitySchedule` and `CanUseRevenueSchedule`, are restricted picklist fields and are available only if the org has the schedules feature enabled.

Usage

Use this object to define the default product information for your org. This object is associated by reference with `Pricebook2` objects via `PricebookEntry` objects. The same product can be represented in different price books as price book entries. In fact, the same product can be represented multiple times (as separate `PricebookEntry` records) in the same price book with different prices or currencies. A

product can only have one price for a given currency within the same price book. To be used in custom price books, all standard prices must be added as price book entries to the standard price book.

You can query the products that have been configured for your org. For example, you can allow your client application to obtain valid product IDs for use when configuring PricebookEntry records via the API. Your client application can perform the following tasks on PricebookEntry objects:

- Query
- Create for the standard pricebook or custom pricebooks.
- Update
- Delete
- Change the `IsActive` field when creating or updating records

This object is defined only for those orgs that have products enabled as a feature. If the org does not have the products feature, this object does not appear in the `describeGlobal()` call, and you can't describe or query this object.

If you try to delete a product via the API but there is an opportunity that uses that product, the delete fails. The workaround is to delete the product in the user interface, which gives you an option to archive the product.

 **Note:** On opportunities and opportunity products, the workflow rules, validation rules, and Apex triggers fire when an update to a child opportunity product or schedule causes an update to the parent record. This means your custom application logic is enforced when there are updates to the parent record, ensuring higher data quality and compliance with your organization's business policies.

SEE ALSO:

[Object Basics](#)

Product2Feed

Represents a single feed item in the feed displayed on the detail page for a product record. This object is available in API version 18.0 and later.

A product feed shows recent changes to a product record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to products in Salesforce.

 **Note:** Products are represented by Product2 objects. As of API version 8.0, the Product object is no longer available. Requests containing Product are refused, and responses do not contain the Product object.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Product2 object
- "Moderate Chatter"



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of Product2Feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p>

Field	Details
	<p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p>

Field	Details
	<ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the product record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>

Field	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL. PollPost—a poll posted on a feed. ProfileSkillPost—generated when a skill is added to a user's Chatter profile. QuestionPost—generated when a user posts a question. ReplyPost—generated when Chatter Answers posts a reply. RipplePost—generated when a user creates a Thanks badge in Work.com.

Field	Details
	<ul style="list-style-type: none"> • <code>TextPost</code>—a direct text entry on a feed. • <code>TrackedChange</code>—a change or group of changes to a tracked field. • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a product record.

SEE ALSO:

[Product2](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

ProductEntitlementTemplate

Represents predefined terms of customer support (Entitlement) that users can add to products (Product2).

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
EntitlementTemplateId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the entitlement template. Must be a valid ID.</p>
Product2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the Product2 associated with the entitlement template. Must be a valid ID.</p>

Usage

Use to query and manage entitlement templates.

SEE ALSO:

[Entitlement](#)

Profile

Represents a profile, which defines a set of permissions to perform different operations, such as querying, adding, updating, or deleting information.

Supported Calls

```
describeSObjects(), getDeleted(), getUpdated(), query(), create(), retrieve(), search(),  
update() delete()
```

Special Access Rules

- Customer Portal users can't access this object.
- Partner portal users can't access this object.

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Description of the profile.</p>
IsSsoEnabled	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, users assigned to this profile can delegate username and password authentication to a corporate database instead of the user database.</p>
LastReferencedDate	<p>Type dateTime</p>

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this profile. Available in API version 29.0 and later.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed this profile. Available in API version 29.0 and later.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The name of the profile.</p>
PermissionsShowCompanyNameAsUserBadge	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When on, a user's company name, if available, will be displayed in place of the community role.</p>
PermissionsPermissionName	<p>Type boolean</p> <p>Properties Filter, Update</p> <p>Description One field for each permission. If true, users assigned to this profile have the named permission. The number of fields varies depending on the permissions for the organization and license type.</p> <p> Tip: To get a list of available permissions in the SOAP API, use <code>describeSObjects()</code>.</p>
UserLicenseId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the UserLicense associated with this profile.</p>
UserType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The category of user license. Each <code>UserType</code> is associated with one or more <code>UserLicense</code> records. Each <code>UserLicense</code> is associated with one or more profiles. In API version 10.0 and later, valid values include:</p> <ul style="list-style-type: none"> • Standard: user license. This user type also includes Salesforce Platform and Salesforce Platform One user licenses. Label is Standard. • PowerPartner: User whose access is limited because he or she is a partner and typically accesses the application through a partner portal or community. Label is Partner. • CSPLitePortal: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal or community. Label is High Volume Portal. • CustomerSuccess: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal User. • PowerCustomerSuccess: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal Manager. <p>Users with this license type can view and edit data they directly own or data owned by or shared with users below them in the Customer Portal role hierarchy.</p> <ul style="list-style-type: none"> • CsnOnly: user whose access to the application is limited to Chatter. This user type includes Chatter Free and Chatter moderator users. Label is Chatter Free. <p><code>UserType</code> replaces <code>LicenseType</code>, which is unavailable as of API version 10.0. In API versions 8.0 and 9.0 <code>LicenseType</code> is still available with the following valid values:</p> <ul style="list-style-type: none"> • AUL: Force.com user license. Label is Apex Platform. • AUL1: Force.com user license with only one user. Label is Apex Platform One. • Salesforce: Salesforce user license. Label is Salesforce. • PackageManager: user who can create and work with managed packages for Force.com AppExchange. Label is Package Manager. • PRM: user whose access is limited because he or she is a partner and typically accesses the application through a partner portal. Label is Partner. • CustomerUser: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal User.

Field	Details
	<ul style="list-style-type: none"> CustomerManager: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal Manager. <p>Users with this license type can view and edit data they directly own or data owned by or shared with users below them in the Customer Portal role hierarchy.</p>

Usage

Use the Profile object to query the set of currently configured user profiles in your organization. Your client application can use Profile objects to obtain valid profile IDs for use when querying or modifying users through the API.

In the user interface, profiles can be used to assign user licenses from specific pools (Force.com Platform user license or Salesforce user license, for example). If a user is assigned to a profile with a different license type, the number of available licenses in the old license type pool increases, one per user changed, and decreases by the same amount in the new license type pool.

SEE ALSO:

[Object Basics](#)
[PermissionSet](#)

ProfileSkill

Represents a profile skill, which describes a user's professional knowledge. This is a global record for the organization, and users are associated through the ProfileSkillUser object.

 **Note:** For information about Live Agent skills, see the [Skill](#) topic.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Description of the profile skill.</p>

Field Name	Details
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp indicating when the current user last viewed a record related to this profile skill. Available in API version 29.0 and later.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp indicating when the current user last viewed this profile skill. Available in API version 29.0 and later.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The name of the profile skill.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The owner of the profile skill.</p>
UserCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of users with the profile skill.</p>

Usage

Use the ProfileSkill object to look up the attributes of a skill that can be assigned to a user. This is a global object and is not owned by any specific user.

ProfileSkillEndorsement

Represents a detail relationship of ProfileSkillUser. An endorsement of a profile skill shows approval and support of another user's publicly declared skill.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The name of the profile skill being endorsed.
ProfileSkillUserId	Type reference Properties Create, Filter, Group, Sort Description The ID of the ProfileSkillUser record that is being endorsed.
UserId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The user ID of the person giving the endorsement.

Usage

Use the ProfileSkillEndorsement object to query about a single endorsement given to a user about a specific skill. Users can't endorse themselves, they can only be endorsed by others unless they are administrators with the "Modify All Data" permission.

ProfileSkillEndorsementFeed

Represents a single feed item in the feed displayed on the detail page for a ProfileSkillEndorsement record. This object is available in API version 34.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The numbers of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	Description
	Available in API version 36.0 and earlier only. The description of the file specified in ContentData.
ContentFileName	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.
ContentSize	Type
	int
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.
ContentType	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then InsertedById is set to the ID of the logged-in user.
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort

Field Name	Details
	<p>Description</p> <p>Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type</p> <p>url</p> <p>Properties</p> <p>Nillable, Sort</p> <p>Description</p> <p>The URL of a LinkPost.</p>
ParentId	<p>Type</p> <p>reference</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p>
	<p>Description ID of the ProfileSkillEndorsement record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type string</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p>
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API.</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p>
	<ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval.

Field Name	Details
	<ul style="list-style-type: none"> • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated.

ProfileSkillEndorsementHistory

Represents the history of changes to the fields of a ProfileSkillEndorsement.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	Type
	picklist

Field Name	Details
	<p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The profile skill endorsement changed field.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the profile skill endorsement field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The old value of the profile skill endorsement field.</p>
ProfileSkillEndorsementId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the profile skill endorsement.</p>

Usage

Use this read-only object to identify changes to a ProfileSkillEndorsement.

ProfileSkillFeed

Represents a single feed item in the feed displayed on the detail page for a profile skill.

A ProfileSkill feed shows recent changes to a ProfileSkill record for any fields that are tracked in feeds, and comments and posts about the record. It is a useful way to stay up-to-date with changes to solutions.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the ProfileSkillFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	Description
	Available in API version 36.0 and earlier only. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code> .
ContentSize	Type int
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. This field is the size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.
ContentType	Type string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. This field is the MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.
InsertedById	Type reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.
IsRichText	Type boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.
	Rich text supports the following HTML tags:
	<ul style="list-style-type: none"> • <p>

Field Name	Details
	<p>Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p>Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nullable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the object type to which the FeedItem object is related. For example, set this field to a <code>UserId</code> to post to someone's profile feed, or an <code>AccountId</code> to post to a specific account.</p>

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. For Work.com thanks posts, it's the ID of the WorkThanks object associated with a RypplePost. This field is typically null for all posts except ContentPost and RypplePost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of ProfileSkillFeed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used.

Field Name	Details
	<ul style="list-style-type: none"> • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • AttachArticleEvent—generated event when a user attaches an article to a case. • CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • CaseCommentPost—generated event when a user adds a case comment for a case object. • ChangeStatusPost—generated event when a user changes the status of a case. • ChatTranscriptPost—generated event when Live Agent transcript is saved to a case. • EmailMessageEvent—generated event when an email related to a case object is sent or received. • FacebookPost—generated when a Facebook post is created from a case. Deprecated. • MilestoneEvent—generated when a case milestone is completed or reaches violation status. • SocialPost—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Usage

Use this object to track changes for a ProfileSkill record.

ProfileSkillHistory

Represents the history of changes to the fields of a ProfileSkill.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ProfileSkillId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. ID of the ProfileSkill.</p>

Usage

Use this read-only object to identify changes to a ProfileSkill.

ProfileSkillShare

Represents a sharing entry on a ProfileSkill.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the ProfileSkill. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All (This value is not valid for <code>create()</code> or <code>update()</code> calls.) <p>This value must be set to an access level that is higher than the organization's default access level for ProfileSkill objects.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the parent object, if any.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description Reason that this sharing entry exists.</p> <p>You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Values may include:</p> <ul style="list-style-type: none"> • Manual—The User or Group has access because a user with “All” access manually shared the ProfileSkill with them. • Owner—The User is the owner of the ProfileSkill or is in a role above the ProfileSkill owner in the role hierarchy.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the ProfileSkill.</p>

Usage

This object is read only. It is visible because of constraints to the ProfileSkill object, but it is ignored and does not control which users and groups can view and edit ProfileSkill records owned by other users.

ProfileSkillUser

Represents a detail relationship of User. The object connects profile skills with users.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
EndorsementCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	<p>Description The number of endorsements.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the skill user.</p>
ProfileSkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the ProfileSkill.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the user. This field can't be changed once it is created.</p>

Usage

Use this object to assign specific skills to specific users. ProfileSkillUser appears on the Overview tab on the Chatter profile page. Users can only create a skill mapping for themselves, they can't create skill mappings for others unless they are administrators with the "Modify All Data" permission. Additionally, users can only edit this object if they are the context user and are not editing the `UserId` field.

ProfileSkillUserFeed

Represents a single feed item in the feed displayed on the detail page for a ProfileSkillUser record. This object is available in API version 34.0 and later.

Supported Calls

```
delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()
```

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The numbers of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then <code>InsertedById</code> is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p>

Field Name	Details
	<ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p>
	 Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the ProfileSkillUser record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>

Field Name	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed items directly from the API.</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file.

Field Name	Details
	<ul style="list-style-type: none"> • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated.

ProfileSkillUserHistory

Represents the history of changes to the fields of a ProfileSkillUser.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p>

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
ProfileSkillUserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. ID of the ProfileSkillUser.</p>

Usage

Use this read-only object to identify changes to a ProfileSkillUser.

PushTopic

Represents a query that is the basis for notifying listeners of changes to records in an organization. This is available from API version 21.0 or later.

Supported Calls

REST: DELETE, GET, PATCH, POST (query requests are specified in the URI)

SOAP: `create()`, `delete()`, `describe()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`

Special Access Rules

- This object is only available if Streaming API is enabled for your organization.
- Only users with “Create” permission can create this record.

Fields

Field	Field Type	Description
ApiVersion	double	<p>Required. API version to use for executing the query specified in <code>Query</code>. It must be an API version greater than 20.0. If your query applies to a custom object from a package, this value must match the package's <code>ApiVersion</code>.</p> <p>Example value: 38.0</p> <p>Field Properties: Create, Filter, Sort, Update</p>
Description	string	<p>Description of the PushTopic. Limit: 400 characters</p> <p>Field Properties: Create, Filter, Sort, Update</p>
ID	ID	<p>System field: Globally unique string that identifies a record.</p> <p>Field Properties: Default on create, Filter, Group, idLookup, Sort</p>
isActive	boolean	<p>Indicates whether the record currently counts towards the organization's limit.</p> <p>Field Properties: Create, Default on create, Filter, Group, Sort, Update</p>
IsDeleted	boolean	<p>System field: Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p> <p>Field Properties: Default on create, Filter, Group, Sort</p>
Name	string	<p>Required. Descriptive name of the PushTopic, such as <code>MyNewCases</code> or <code>TeamUpdatedContacts</code>. Limit: 25 characters. This value identifies the channel and must be unique.</p> <p>Field Properties: Create, Filter, Group, Sort, Update</p>
NotifyForFields	picklist	<p>Specifies which fields are evaluated to generate a notification.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • All • Referenced (default) • Select • Where <p>Field Properties: Create, Filter, Sort, Update</p>
NotifyForOperations	picklist	<p>Specifies which record events may generate a notification.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • All (default) • Create • Extended • Update <p>Field Properties for API version 28.0 and earlier: Create, Filter, Sort, Update</p>

Field	Field Type	Description
Field Properties for API version 29.0 and later: Filter, Sort		
		In API version 29.0 and later, this field is read-only, and will not contain information about delete and undelete events. Use <code>NotifyForOperationCreate</code> , <code>NotifyForOperationDelete</code> , <code>NotifyForOperationUndelete</code> and <code>NotifyForOperationUpdate</code> to specify which record events should generate a notification.
		A value of <code>Extended</code> means that neither create or update operations are set to generate events.
<code>NotifyForOperationCreate</code>	boolean	<code>true</code> if a create operation should generate a notification, otherwise, <code>false</code> . Defaults to <code>true</code> . This field is available in API version 29.0 and later.
<code>NotifyForOperationDelete</code>	boolean	<code>true</code> if a delete operation should generate a notification, otherwise, <code>false</code> . Defaults to <code>true</code> . Clients must connect using the <code>cometd/29.0</code> (or later) Streaming API endpoint to receive delete and undelete event notifications. This field is available in API version 29.0 and later.
<code>NotifyForOperationUndelete</code>	boolean	<code>true</code> if an undelete operation should generate a notification, otherwise, <code>false</code> . Defaults to <code>true</code> . Clients must connect using the <code>cometd/29.0</code> (or later) Streaming API endpoint to receive delete and undelete event notifications. This field is available in API version 29.0 and later.
<code>NotifyForOperationUpdate</code>	boolean	<code>true</code> if an update operation should generate a notification, otherwise, <code>false</code> . Defaults to <code>true</code> . This field is available in API version 29.0 and later.
Query	string	<p>Required. The SOQL query statement that determines which record changes trigger events to be sent to the channel.</p> <p>Limit: 1,300 characters</p> <p>Field Properties: Create, Filter, Sort, Update</p>

PushTopic and Notifications

The PushTopic defines when notifications are generated in the channel. This is specified by configuring the following PushTopic fields:

- PushTopic Queries
- Events
- Notifications

QuantityForecast

Represents a quantity-based forecast.

The API also provides revenue-based forecasts using RevenueForecast.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- Requires the “View All Data” permission.
- Customer Portal users can’t access this object.

Fields

Field	Details
Closed	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only. A rollup of opportunities or opportunity line items that have closed in this period.</p>
Commit	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The owner's Commit total.</p>
CommitComment	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Read-only. The comment entered when the owner edited his or her Commit total from the Adjusted Total link on the forecast edit page.</p>
CommitOverride	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Read-only. The owner's override of their own My Commit total.</p>

Field	Details
DefaultRollupCommit	Type double
	Properties Filter, Nillable, Sort
	Description Read-only. The owner's standard Commit rollup, including their own opportunities and forecast-level overrides from subordinate users in the role hierarchy.
DefaultRollupUpside	Type double
	Properties Filter, Nillable, Sort
	Description Read-only. The owner's standard Best Case rollup, including their own opportunities and forecast-level overrides from subordinate users in the role hierarchy.
InvalidationDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description Read-only. If not blank, indicates that the rollup numbers for Forecast Override fields that represent calculated (summarized) amounts may not be up to date.
ManagerChoiceCommit	Type picklist
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Read-only. The manager's choice regarding the commit amount:
	<ul style="list-style-type: none"> • DefaultRollup: Use the manager's default commit rollup for owner's forecast, which reflects the manager's opportunity forecast overrides. • AcceptForecast (default selection): Accept the forecast owner's Adjusted Total commit amount, which may or may not be an override. • ManagerManualOverride: Use the manager's manual override. • OpportunityOnlyRollup: Use the opportunity rollup, including opportunity forecast overrides, but excluding any forecast-level (Adjusted Total) overrides.
ManagerChoiceUpside	Type picklist
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	<p>Description Read-only. The manager's choice regarding the best case amount:</p> <ul style="list-style-type: none"> • DefaultRollup: Use the manager's default best case rollup for owner's forecast, which reflects the manager's opportunity forecast overrides. • AcceptForecast (default selection): Accept the forecast owner's Adjusted Total best case amount, which may or may not be an override. • ManagerManualOverride: Use the manager's manual override. • OpportunityOnlyRollup: Use the opportunity rollup, including opportunity forecast overrides, but excluding any forecast-level (Adjusted Total) overrides.
ManagerClosed	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only. The manager's closed total for the owner's forecast, including any opportunity or opportunity product overrides made by the manager.</p>
ManagerCommit	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The manager's Commit total.</p>
ManagerCommitOverride	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Read-only. The manager's manual override of the forecast owner's Commit total. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.</p>
ManagerDefaultRollupCommit	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only. The manager's standard Commit rollup for the forecast owner. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.</p>

Field	Details
ManagerDefaultRollupUpside	Type double
	Properties Filter, Nillable, Sort
	Description Read-only. The manager's standard Best Case rollup for the forecast owner. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerId	Type reference
	Properties Filter, Group, Nillable, Sort
	Description Read-only. ID of the direct manager of the user who owns this forecast.
ManagerOpportunityRollupCommit	Type double
	Properties Filter, Nillable, Sort
	Description Read-only. The manager's view of the forecast owner's opportunity-level Commit rollup, ignoring all forecast overrides. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerOpportunityRollupUpside	Type double
	Properties Filter, Nillable, Sort
	Description Read-only. The manager's view of the forecast owner's opportunity-level Best Case rollup, ignoring all forecast overrides. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerPipeline	Type double
	Properties Filter, Nillable, Sort

Field	Details
	<p>Description The manager's pipeline total for the owner's forecast, including any opportunity or opportunity product overrides made by the manager.</p>
ManagerUpside	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The manager's Best Case total.</p>
ManagerUpsideOverride	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The manager's manual override of the forecast owner's Best Case total. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.</p>
OpportunityRollupClosed	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The owner's Closed total for his or her opportunities only.</p>
OpportunityRollupCommit	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The owner's Commit total for his or her opportunities only.</p>
OpportunityRollupPipeline	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The owner's Pipeline total for his or her opportunities only.</p>

Field	Details
OpportunityRollupUpside	Type double Properties Filter, Nillable, Sort
	Description The owner's Best Case total for his or her opportunities only.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort
	Description ID of the User who owns this forecast. Required on create.
PeriodId	Type reference Properties Filter, Nillable, Group, Sort
	Description The ID of the Period that contains the <code>StartDate</code> .
Pipeline	Type double Properties Filter, Nillable, Sort
	Description The total pipeline rollup from subordinates in the role hierarchy, including the owner's opportunities.
ProductFamily	Type picklist Properties Create, Filter, Group, Nillable, Sort
	Description The value chosen in the Product Family picklist, which can be configured from the object management settings for products. This field is relevant if you have chosen Product Families as the Forecast Type in Forecasts Settings. If you are not forecasting by product family or if the forecast represents opportunities that are not associated with a product family, then this field is blank. Required on create.
Quota	Type double

Field	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The quota amount for the period. You can create, update, upsert, and delete. Requires the “Modify All Data” and “Manage Users” permission. Required on create.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The start date of this forecast. The ID of the Period that contains this date is written to the <code>PeriodId</code> field if it changes. A new Period is created if none exists. Required on create.</p>
Upside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The owner's Best Case total.</p>
UpsideComment	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description The comment entered when the owner edited his or her Best Case total.</p>
UpsideOverride	<p>Type double</p> <p>Properties Filter, Nillable, Update</p> <p>Description Read only. The owner's override of their own My Best Case total.</p>

Usage

Query this object to support customizable forecasts based on quantities.

You can update one field, `Quota`, which means that you can mass update sales users' quotas instead of updating them one by one in the user interface. Editing `Quota`, which means that you can mass update sales users' quotas instead of updating them one by one in the user interface. Editing `Quota` requires the "Modify All Data" and "Manage Users" permissions.

The rollup fields always reflect opportunity and opportunity product overrides by the forecast owner or one of the forecast owner's subordinates in the role hierarchy. In addition, the manager rollup fields include overrides by the forecast owner's direct manager in the role hierarchy.

Some of the rollup fields ignore forecast-level (**Adjusted Total**) overrides, but they never ignore opportunity forecast overrides that are visible to the owner or manager.

SEE ALSO:

[RevenueForecast](#)

[QuantityForecastHistory](#)

QuantityForecastHistory

Represents historical information about quantity-based forecasts that have been submitted (saved) in the user interface.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

- Requires the "View All Data" permission.
- Customer Portal users can't access this object.

Fields

Field	Details
Closed	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The closed amount of the forecast.</p>
Commit	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The commit amount of the forecast.</p>

Field	Details
CommitComments	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Comments about the commit value.</p>
CommitOverridden	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the commit value was overridden (<code>true</code>) or not (<code>false</code>).</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
ForecastOverrideId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the related forecast override.</p>
Pipeline	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The pipeline amount of the forecast.</p>
Quota	<p>Type currency</p> <p>Properties Filter, Nillable</p>

Field	Details
	<p>Description The quota amount of the forecast.</p>
Upside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The best case amount of the forecast.</p>
UpsideComments	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description Comments about the upside value.</p>
UpsideOverridden	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the upside value was overridden (<code>true</code>) or not (<code>false</code>).</p>

Usage

This is a read-only object specific to customizable forecasting.

When a user submits a revenue-based forecast in the user interface, a new record is created. If the same forecast is ever resubmitted, additional records are added. The `CreatedDate` of a record reflects the day on which the forecast was submitted. This object respects field level security on the parent object.

SEE ALSO:

[QuantityForecast](#)
[RevenueForecastHistory](#)

QueueRoutingConfig

Represents the settings that determine how work items are routed to agents. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
CapacityPercentage	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel. For example, you might give phone calls a capacity percentage of <code>100</code>. If an agent receives a phone call, the agent won't receive new work items until the call ends, because at that point the agent's capacity will have reached 100%. This field is available in API version 33.0 and later.</p>
CapacityWeight	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel. For example, if an agent has a capacity of <code>6</code>, and cases are assigned a capacity weight of <code>2</code>, an agent can be assigned up to 3 cases before the agent is at capacity and can't receive new work items. This field is available in API version 33.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the presence status.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the presence status.</p>
PushTimeout	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The number of seconds set for push timeout. 0 is returned when push timeout isn't enabled. Available in API version 36.0 and later.</p>
RoutingModel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The routing type that determines how work items are routed (pushed) to agents. Possible values are <code>Least Active</code> and <code>Most Available</code>.</p>
RoutingPriority	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The priority in which work items from the service channels that are related to this routing configuration are routed to agents. Work items from routing configurations that have lower priority values (for example, <i>0</i>) are routed to agents first.</p>

Field	Details
ServiceChannelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the service channel that's associated with this configuration.</p>

Question

Represents a question in a community that users can view and reply to.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Field	Details
BestReplyId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the Reply that has been identified as the best answer to the question. You must use the user interface to identify the best answer for a question.</p>
BestReplySelectedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the User who selected the best answer to the question. This field is available in API version 24.0 and later. In API version 24.0 through version 29.0, you must update this field using the UI. In API version 38.0 and later, you can update this field using the API.</p>
Body	<p>Type textarea</p>

Field	Details
	Properties
	Create, Nullable, Update
	Description
	Description of the question.
CommunityId	Type
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	The zone ID associated with the question. Once you create a question, you can't change the zone ID associated with that question.
CreatorFullPhotoUrl	Type
	string
	Properties
	Filter, Group, Nullable, Sort
	Description
	URL of the user's profile photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.
CreatorName	Type
	string
	Properties
	Filter, Group, Nullable, Sort
	Description
	Name of the user who posted the question or reply. Only the first name of internal users (agents) appears to portal users in the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.
CreatorSmallPhotoUrl	Type
	string
	Properties
	Filter, Group, Nullable, Sort
	Description
	URL of the user's thumbnail photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.
HasSingleFieldForContent	Type
	boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the content of a Chatter Answers question is:</p> <ul style="list-style-type: none">• Included in only one field: <code>Title</code> if the content is unformatted and less than 255 characters; or <code>Body</code> if the content is formatted or more than 255 characters (<code>true</code>)• Included in two fields: <code>Title</code> and <code>Body</code> (<code>false</code>) <p>This field also determines if content displays in one or two fields in Chatter Answers question feeds.</p> <p>This field is available in API version 25.0 and later.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastReplyDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time the last reply (child Reply object) was posted.</p>
LastReplyId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read only. The ID of the last reply (child Reply object) posted to the question.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>

Field	Details
MostReportAbusesOnReply	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The most number of user-reported abuses on a Reply associated with the question. This field is available in API version 24.0 and later.</p>
NumReplies	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of replies (child Reply object) that users have submitted for the question.</p>
NumReportAbuses	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the number of user-reported abuses on the question. This field is available in API version 24.0 and later.</p>
NumSubscriptions	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the number of users following the question. This field is available in API version 24.0 and later.</p>
Origin	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The source of the question, such as Chatter Answers. This field is available in API version 24.0 and later.</p>
Title	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The descriptive title of the question.</p>
UpVotes	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The total number of up votes for the question.</p>
VoteScore	<p>Type double</p> <p>Properties Filter , Nillable, Sort</p> <p>Description The internal score of the question, used to sort questions and articles on the Popular tab in the application user interface. The internal algorithm that determines the score gives older votes less weight than newer votes, simulating exponential decay. The score itself does not display in the application user interface.</p> <p> Note: Unlike other fields of type double, you can't use a SOQL aggregate function with this field.</p>

Usage

Use this object to track questions in azone.

QuestionDataCategorySelection

A data category selection represents a data category that classifies a question.

This object can be used to associate a question with a data category from a data category group or to query the categorization for a question.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

To create, read or update data category selection, you must have create, read or update permission on the categorized question. Users who can update question can also delete its category selection. Users who can create questions can only select categories visible to their role.

Fields

Field Name	Details
DataCategoryGroupName	<p>Type DataCategoryGroupReference</p> <p>Properties Create</p> <p>Description Unique name of the data category group which has a category associated with the question.</p>
DataCategoryName	<p>Type DataCategoryGroupReference</p> <p>Properties Create</p> <p>Description Unique name of the data category associated with the question.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the question associated with the data category selection.</p>

Usage

Every question can be categorized in a data category. You can use the QuestionDataCategorySelection object to query and manage question categorization. Client applications can create categorization for a question. They can also delete, query, and retrieve question categorization.



Warning: Even though the API lets you select more than one category for QuestionDataCategorySelection, the Answers tab only supports one data category selection for questions. Selecting multiple categories through QuestionDataCategorySelection may result in unexpected behavior in the Answers tab, such as losing your multiple selections. You should only select one data category when using QuestionDataCategorySelection.

Sample Code—Java

In the following example, the `selectCategory` method adds a category to a question data category selection. The `retrieveCategorySelections` method returns all the categories from a question data category selection.

```
public void selectCategory(ID parentId, String categoryGroupName, String categoryName) {
    try {
        QuestionDataCategorySelection categorySelection = new
QuestionDataCategorySelection();

        categorySelection.setParentId(parentId);
        categorySelection.setDataCategoryGroupName(categoryGroupName);
        categorySelection.setDataCategoryName(categoryName);
        binding.create(new SObject[]{categorySelection});
    } catch (RemoteException e) {
        System.out.println("An unexpected error has occurred." + e.getMessage());
    }
}

public String[] retrieveCategorySelections(String parentId) {
    QueryResult qr = null;

    try {
        qr = binding.query("SELECT DataCategoryName FROM QuestionDataCategorySelection
WHERE Id = '" + parentId + "'");
    } catch (RemoteException e) {
        System.out.println("An unexpected error has occurred." + e.getMessage());
    }

    String[] categoryNames = new String[qr.getRecords().length];
    for (int index = 0; index < qr.getRecords().length; index++) {
        categoryNames[index] =
((QuestionDataCategorySelection)qr.getRecords()[index]).getDataCategoryName();
    }

    return categoryNames;
}
```

Salesforce Knowledge uses a similar object for article data category selection. See [Article Type__DataCategorySelection](#) for SOQL examples using this object.

SEE ALSO:

[Article Type__DataCategorySelection](#)

[Data Categories](#)

QuestionReportAbuse

Represents a user-reported abuse on a Question in a Chatter Answers community. This object is available in API version 24.0 and later.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the Question from which the user reported abuse.</p>
QuestionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the Question from which the user reported abuse.</p>
Reason	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The reason the user reported abuse on the Question, such as <code>Spam</code>, <code>Hateful</code>, or <code>Inappropriate</code>.</p>

Usage

Use this object to track user-reported abuse on questions created in a Chatter Answers community.

QuestionSubscription

Represents a subscription for a user following a Question. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field	Details
CommunityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the zone associated with the Question the user is following. This field can't be updated.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The name of the question subscription.</p>
QuestionCreatedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Required. Creation date of the Question which the user is following. This field can't be updated.</p>
QuestionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. ID of the Question which the user is following. This field can't be updated.</p>
SubscriberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. ID of the User who is following the Question. This field can't be updated.</p>

Usage

Things to consider when following a Question:

- A user can only follow questions that they have permission to view.
- Administrators and users with the “Modify All Data” permission can configure other users to follow questions that the other user has read access to.
- Administrators and users with the “Modify All Data” permission can configure users to stop following questions.

Queries on QuestionSubscription:

- Users with the “Read” permission on Question can see which questions other users are following.
- A query must include a LIMIT clause and the limit can’t exceed 1,000.
- A query using a WHERE clause can only filter by fields on Question.

QueueSobject

Represents the mapping between a queue Group and the sObject types associated with the queue, including custom objects.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object. A queue is a Group whose Type is Queue. To create a Group, you must have the “Manage Users” permission.

Fields

Field	Details
QueueId	Type reference Properties Create, Filter, Group, Sort Description The ID of a queue.
SObjectType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description A list of object types that can be associated with the queue specified by the QueueId.

Usage

Use this object to associate a queue with the sObject that can be associated with the queue, including custom objects.

SEE ALSO:

[Object Basics](#)

QuickText

This object stores a snippet of text that allows an agent to send a quick response to a customer in the Live Agent console. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
Category	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description A customizable picklist that can be used to group multiple related Quick Text records together</p>
Channel	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Update</p> <p>Description A multi-select picklist that can be used to specify where specific Quick Text messages should be available, such as in Live Agent or in the Email publisher in Case Feed.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Message	<p>Type textarea</p> <p>Properties Create, Filter (unavailable in API version 25.0 and later), Sort (unavailable in API version 25.0 and later), Update</p> <p>Description The content of the Quick Text record</p>
Name	<p>Type string</p> <p>Properties Create, Filter (unavailable in API version 25.0 and later), Group, Sort (unavailable in API version 25.0 and later), Update</p> <p>Description A descriptive label for the Quick Text record</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the User who owns the Quick Text record</p>

Usage

Use this object to create and manage the Quick Text messages available to agents in the Live Agent console. You can categorize multiple QuickText records into groups using the Category field. The Category field can also be a parent to multiple custom dependent Picklist fields to create a hierarchical structure of categories.

QuickTextHistory

Represents changes to field values on a QuickText object. This object is available in API version 24.0 and later.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The old value of the field that was changed</p>
QuickTextId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the QuickText that was changed</p>

Usage

Use this object to identify changes to a QuickText.

QuickTextOwnerSharingRule

Represents a rule for sharing a QuickText object with users other than the owner.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`,
`update()`, `upsert()`

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Group, Sort, Update</p> <p>Description A value that represents the type of sharing allowed. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with</p>

Field Name	Details
	a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.
	 Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.
GroupID	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. QuickText objects owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the user interface.</p>
UserorGroupID	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description The ID representing the target user or group. Target users or groups have access to the QuickText.</p>

Usage

Use this object to manage the sharing rules for a QuickText object.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

QuickTextShare

Represents a sharing entry on a QuickText object. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the QuickText. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit• All (This value is not valid for <code>create()</code> or <code>update()</code> calls.) <p>This value must be set to an access level that is higher than the organization's default access level for QuickText objects.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the parent object, if any</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>

Field Name	Details
	<p>Values may include:</p> <ul style="list-style-type: none"> • Manual—The User or Group has access because a user with “All” access manually shared the QuickText with them. • Owner—The User is the owner of the QuickText or is in a role above the QuickText owner in the role hierarchy.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the QuickText</p>

Usage

This object lets you determine which users and groups can view and edit QuickText records owned by other users.

If you attempt to create a new record that matches an existing record, the `create()` call updates any modified fields and returns the existing record.

Quote

The Quote object represents a quote, which is a record showing proposed prices for products and services. Available in API version 18.0 and later.

Quotes can be created from and synced with opportunities, and emailed as PDFs to customers

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description ID of the account that’s associated with the quote.</p>

Field	Details
AdditionalAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description Compound form of the additional address. Read-only. See Address Compound Fields for details on compound address fields.</p>
AdditionalCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description City for the quote's additional address. Up to 40 characters allowed.</p>
AdditionalCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Country for the quote's additional address. Up to 80 characters allowed.</p>
AdditionalCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO country code for the quote's additional address.</p>
AdditionalLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with AdditionalLongitude to specify the precise geolocation of an additional address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>
AdditionalLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description Used with AdditionalLatitude to specify the precise geolocation of an additional address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
AdditionalName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name associated with the quote's additional address. Limited: 255 characters.</p>
AdditionalPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal Code for the quote's additional address.</p>
AdditionalState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the quote's additional address. Up to 80 characters allowed.</p>
AdditionalStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO state code for the quote's additional address.</p>
AdditionalStreet	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Group, Sort, Update</p> <p>Description Street name for the quote's additional address.</p>
BillingAddress	<p>Type address</p>

Field	Details
	<p>Properties Filter, Nillable</p> <p>Description Compound form of the billing address. Read-only. See Address Compound Fields for details on compound address fields.</p>
BillingCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description City for the quote's billing address. Up to 40 characters allowed.</p>
BillingCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Country for the quote's billing address. Up to 80 characters allowed.</p>
BillingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO country code for the quote's billing address.</p>
BillingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>
BillingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
BillingName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Entity that the quote is billed to.</p>
BillingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal Code for the quote's billing address.</p>
BillingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the quote's billing address. Up to 80 characters allowed.</p>
BillingStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO state code for the quote's billing address.</p>
BillingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street name for the quote's billing address.</p>
ContactId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the contact that's associated with the quote.</p>
ContractId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the contract that's associated with the quote.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Restricted picklist</p> <p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. If the organization has multicurrency and a <code>Pricebook2Id</code> specified on the quote, then the currency value of this field must match the currency of the <code>PricebookEntry</code> objects that are associated with any quote line items it has. This value is copied from the related Opportunity and can't be changed.</p>
Description	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description Text description of the quote. Limit: 32,000 characters.</p>
Discount	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Difference between the sum of the <code>QuoteLineItem</code> record's <code>Subtotal</code> and the sum of the <code>QuoteLineItem</code> record's <code>Discount</code> totals. Expressed as a percentage.</p>
Email	<p>Type email</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The email address of the contact who's associated with the quote.</p>
ExpirationDate	<p>Type date</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The date when this quote is no longer valid.</p>
Fax	<p>Type phone</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The fax number for the contact who's associated with the quote.</p>
GrandTotal	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The total price of the quote plus shipping and taxes.</p>
IsSyncing	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the quote is syncing with an opportunity.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>

Field	Details
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
LineItemCount	<p>Type int</p> <p>Properties Filter, Nillable</p> <p>Description The number of line items on the quote.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, idLookup, Update</p> <p>Description Required. Name for the quote. Limit: 225 characters.</p>
OpportunityId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID for the opportunity associated with the quote.</p>
Phone	<p>Type phone</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The phone number of the contact who's associated with the quote.</p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p>

Field	Details
	<p>Description ID of the price book associated with the quote.</p>
QuoteNumber	<p>Type string</p> <p>Properties Defaulted on create, Filter</p> <p>Description A system-generated number that identifies the quote.</p>
QuoteToAddress	<p>Type address</p> <p>Properties Filter, Nullable</p> <p>Description Compound form of the quote to address. Read-only. See Address Compound Fields for details on compound address fields.</p>
QuoteToCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description City for the address to send the quote to for approval, such as a third party-agency representing a buyer. Up to 40 characters allowed.</p>
QuoteToCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Country for the address to send the quote to for approval. Up to 80 characters allowed.</p>
QuoteToLatitude	<p>Type double</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Used with <code>QuoteToLongitude</code> to specify the precise geolocation of a quote to address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>

Field	Details
QuoteToLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with QuoteToLatitude to specify the precise geolocation of a quote to address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
QuoteToName	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The name of the entity (such as a person or business) that the quote is sent to for approval. Limit: 255 characters.</p>
QuoteToPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal code for the address to send the quote to for approval.</p>
QuoteToState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the address to send the quote to for approval. Up to 80 characters allowed.</p>
QuoteToStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street name for the address to send the quote to for approval.</p>
RecordTypeID	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the record type assigned to the object.</p>
ShippingAddress	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description Compound form of the shipping address. Read-only. See Address Compound Fields for details on compound address fields.</p>
ShippingCity	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description City for the quote's shipping address. Up to 40 characters allowed.</p>
ShippingCountry	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Country for the quote's shipping address. Up to 80 characters allowed.</p>
ShippingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO country code for the quote's shipping address.</p>
ShippingHandling	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The total shipping and handling costs for the quote.</p>

Field	Details
ShippingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>
ShippingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
ShippingName	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The name of the entity (such as a person or business) that the quote is sent to for approval.</p>
ShippingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Postal code for the quote's shipping address.</p>
ShippingState	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description State for the quote's shipping address. Up to 80 characters allowed.</p>
ShippingStateCode	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ISO state code for the quote's shipping address.</p>
ShippingStreet	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Street name for the quote's shipping address.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Update</p> <p>Description The status of the quote. The standard options are:</p> <ul style="list-style-type: none">• —None—• Draft• Needs Review• In Review• Approved• Rejected• Presented• Accepted• Denied
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The line item's sales price multiplied by the quantity and minus the discount.</p>
Tax	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Update</p>

Field	Details
	<p>Description The total taxes for the quote.</p>
TotalPrice	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The total of the quote line items after discounts and before taxes and shipping.</p>

Usage

Use Quote to manage proposed product prices for customers. To update a Quote, your client application needs “Edit” permission.

- Client applications can create, update, delete, and query Attachment records associated with a quote via the API.
- You can sync a quote and its parent Opportunity.

SEE ALSO:

[QuoteLineItem](#)

[QuoteDocument](#)

[Opportunity](#)

QuoteDocument

Represents a quote in document format. Available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Restricted picklist</p> <p>Description Available only for organizations with the multicurrency feature enabled.</p>

Field	Details
	Contains the ISO code for any currency allowed by the organization. If the organization has multicurrency and a Pricebook2Id specified on the quote, then the currency value of this field must match the currency of the PricebookEntry objects that are associated with any quote line items it has.
Discount	<p>Type percent</p> <p>Properties Create, Filter, Nillable</p> <p>Description The discount for the quote used in the document.</p>
Document	<p>Type base64</p> <p>Properties Create, Nillable</p> <p>Description The binary data of the document stored in the QuoteDocument object.</p>
GrandTotal	<p>Type currency</p> <p>Properties Create, Filter, Nillable</p> <p>Description Grand total for the quote used in the document.</p>
Name	<p>Type string</p> <p>Properties Filter, idLookup</p> <p>Description Name of the quote document.</p>
QuoteId	<p>Type reference</p> <p>Properties Create, Filter, Sort</p> <p>Description ID for the quote used for the document.</p>
Template	<p>Type string</p>

Field	Details
	Properties Create, Filter
	Description Reserved for future use.

Usage

Use the QuoteDocument object to store a document that can be used to present the quote information to the customer.

SEE ALSO:

[Quote](#)
[QuoteLineItem](#)

QuoteLineItem

The QuoteLineItem object represents a quote line item, which is a member of the list of Product2 products associated with a Quote, along with other information about those line items on that quote. Available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeLayout(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

The user must have "Edit" permissions on Quote records in order to create or update quote line items on a quote. The user must have "Edit" permissions on Quote records to delete a quote line item.

Fields

Field	Details
CurrencyIsoCode	Type picklist Properties Filter, Restricted picklist Description Available only for organizations enabled for multiples currencies. Contains the ISO code for any currency allowed by the organization.

Field	Details
	<p>If the organization has multicurrency and a Pricebook2 is specified on the quote (the <code>Pricebook2Id</code> field is not blank), then the currency value of this field must match the currency of the <code>PricebookEntry</code> objects for any associated quote line items.</p> <p>This value is copied from the related Quote and can't be changed.</p>
<code>Description</code>	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Text description of the line item. Limit: 225 characters.</p>
<code>Discount</code>	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Editable number from 0 to 100.</p>
<code>HasQuantitySchedule</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. Indicates whether the opportunity line item that the quote line item is synced with has a quantity schedule.</p>
<code>HasRevenueSchedule</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. Indicates whether the opportunity line item that the quote line item is synced with has a revenue schedule. If this object has a revenue schedule, the <code>GrandTotal</code> and <code>TotalPrice</code> fields can't be updated. In addition, the <code>Quantity</code> field can't be updated if this object has a quantity schedule. The system ignores any attempt to update this field. The update isn't rejected but the updated value is ignored.</p>
<code>HasSchedule</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description Read-only. Indicates whether the line item uses schedules.</p>
LineNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description Read-only. Automatically-generated number identifying the quote line item. In the form of QL-XXXXX.</p>
ListPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only. Corresponds to the <code>UnitPrice</code> on the PricebookEntry that is associated with this line item, which can be in the standard price book or a custom price book. A client application can use this information to show whether the unit price (or sales price) of the line item differs from the price book entry list price.</p>
PricebookEntryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the associated PricebookEntry. Exists only for those organizations that have Products enabled as a feature.</p>
Product2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description ID of the Product2 associated with this QuoteLineItem.</p>
Quantity	<p>Type double</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The number of units for the line item.</p>

Field	Details
QuoteId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. ID of the associated Quote.</p>
ServiceDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Date when the product revenue will be recognized and the product quantity will be shipped.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The value of where the line item is in the sorted order, such as 1, 2, and so on. The SortOrder value determines the order in which a quote line item displays in the Quote Detail page. Client applications can use this to match the sort order in Salesforce. This field is only available in API versions 21.0 and greater.</p>
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The line item's <code>Quantity</code> multiplied by the <code>UnitPrice</code>.</p>
TotalPrice	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description Read-only. Calculated by applying the <code>Discount</code> to the <code>Subtotal</code>. This field is nillable, but you can't set both <code>TotalPrice</code> and <code>UnitPrice</code> to null in the same update. To insert the <code>TotalPrice</code> for a quote line item via the API (given only a unit price and the quantity), calculate this field as the unit price multiplied by the quantity. This field is read-only if the quote line item has a revenue schedule.</p>

Field	Details
UnitPrice	<p>Type currency</p> <p>Properties Create, Filter, Update</p> <p>Description The price per unit for the quote line item.</p>

Usage

A Quote record can have QuoteLineItem records only if the Quote has a Pricebook2. A QuoteLineItem must correspond to a Product2 that is listed in the quote's Pricebook2.



Note: If the multicurrency option has been enabled, the `CurrencyIsoCode` field is present. It can't be modified, it is always set to the value of the `CurrencyIsoCode` of the parent Quote.

Effects on Quotes

Quotes that have associated QuoteLineItem objects are affected in the following ways:

- Creating a QuoteLineItem increments the Quote value by the `TotalPrice` of the QuoteLineItem.
- When you create or update a QuoteLineItem, the API verifies that the line item corresponds to a PricebookEntry in the Pricebook2 that is associated with the quote.

SEE ALSO:

[Quote](#)
[QuoteDocument](#)
[Opportunity](#)

RecentlyViewed

Represents records that the current user has recently viewed or referenced (by viewing a related record).

Supported Calls

`query()`, `update()`

Special Usage Rules

The RecentlyViewed object does not support the Report, KnowledgeArticle, and Article objects.

Fields

Field	Details
Alias	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The alias on the record.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The email address on the record.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name on the record. If the recently viewed record is a user, this is the user's first name.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description The ID of the recently viewed record.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the recently viewed record is an active user (<code>true</code>) or not (<code>false</code>). This field contains a value only if the recently viewed record is a user.</p>
LastName	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The last name on the record.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name on the recently viewed record. If the recently viewed record is a user, contact, or lead, the value is a concatenation of the <code>firstname</code> and <code>lastname</code> field values.</p>
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the community that this group is part of. This field is available only if Salesforce Communities is enabled in your organization. You can add a <code>NetworkId</code> only when creating a group. You can't change or add a <code>NetworkId</code> for an existing group. This field is available in API version 27.0 and later.</p>
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The phone number on the record.</p>
ProfileId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the recently viewed record is a user, this is the user's profile ID.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the recently viewed record is a user, this is the title of the user; for example CFO or CEO.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The sObject type for this recently viewed record. Valid values include any standard or custom objects that RecentlyViewed supports.</p>
UserRoleId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user role associated with this object.</p>

Usage

This object provides a heterogeneous list of different object types and consists of recently viewed records or records that were recently referenced (a related record was viewed). A record is considered viewed when the user sees the details associated with it, but not when the user sees it in a list with other records. Use this object to programmatically construct a list of recently viewed items specific to the current user, for example, on a custom user interface or for search auto-complete options. You can also retrieve a filtered list of records by object type (**Type**). The RecentlyViewed data is periodically truncated down to 200 records per object.

Use this query in your code to retrieve a list of all the records that were recently viewed. The results are ordered from most to least recent.

```
SELECT Id, Name
FROM RecentlyViewed
WHERE LastViewedDate !=null
ORDER BY LastViewedDate DESC
```

Use this query to retrieve data that was either viewed or referenced, but only for a limited set of objects.

```
SELECT Id, Name
FROM RecentlyViewed
WHERE Type IN ('Account', 'Contact', 'Plan__c')
ORDER BY LastViewedDate DESC
```

This query retrieves a list of all recently viewed contacts with contact-specific fields, such as the contact's account name, and the custom website field. Records are ordered from most to least recent.

```
SELECT Account.Name, Title, Email, Phone, Website__c
FROM Contact
WHERE LastViewedDate != NULL
ORDER BY LastViewedDate DESC
```

RecordType

Represents a record type.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
BusinessProcessId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Required for Opportunity and Lead record types in API version 17.0 and later. ID of an associated BusinessProcess.</p>
Description	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A description of this record. Limit: 255 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Record Type Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether this record is active (<code>true</code>) or not (<code>false</code>). Only active record types can be applied to records. Label is Active.</p>
IsPersonType	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether this record has been designated as a person account (<code>true</code>) or not (<code>false</code>). Visible only if the organization has the person account feature enabled.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Required. Label of the record type in the user interface. Limit: 80 characters. Label is Record Type Label.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>
SObjectType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Object to which this record type applies, including custom objects.</p>

Usage

Use this object to offer different BusinessProcess records and subsets of picklist values to different users based on their Profile. Your client application can describe or query RecordType records.

The following objects have a **RecordTypeId** field:

- Account
- Campaign
- CampaignMember
- Case

- Contact
- Contract
- Lead
- Opportunity
- QuickText
- Solution
- Custom objects

Client applications can create or update values in `RecordTypeId` on these objects, specifying a valid record type ID associated with these objects.

 **Note:** You can't create or update the `RecordTypeId` field on the CampaignMember records. Set the CampaignMember record type using the `CampaignMemberRecordTypeId` field on Campaign.

A client application can retrieve the list of valid record type IDs for a given object by querying the `RecordType`.

SEE ALSO:

[Record Type Objects](#)

RecordTypeLocalization

Represents the translated value of a label for a record type when the Translation Workbench is enabled for your organization.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

- Your organization must be using Professional, Enterprise, Developer, Unlimited, or Performance Edition and be enabled for the Translation Workbench.
- To view this object, you must have the “View Setup and Configuration” permission.

Fields

Field	Details
Language	Type string Properties Create, Filter, Nillable, Restricted picklist Description The language for this translated label.

Field	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Nillable</p> <p>Description The ID of the RecordType associated with the label that is being translated.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The actual translated label for the record type. Label is Translation.</p>

Usage

Use this object to translate the labels of your record types into other supported languages.

Reply

Represents a reply that a user has submitted to a question in an answers community.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description Body of this reply.</p>
CommunityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The zone ID associated with the question and its reply. This field is available in API version 27.0 and later.</p>
CreatorFullPhotoUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description URL of the user's profile photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
CreatorName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Name of the user who posted the question or reply. Only the first name of internal users (agents) appears to portal users in the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later</p>
CreatorSmallPhotoUrl	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description URL of the user's thumbnail photo from the feed. Chatter Answers must be enabled to view this field. This field is available in API version 26.0 and later.</p>
DownVotes	<p>Type int</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The total number of down votes for a reply.</p>
Name	<p>Type string</p>
	<p>Properties Filter, Group, idLookup, Sort</p>
	<p>Description When creating a Reply, the Name field is automatically populated with a truncated, plain text version of the Reply Body field.</p>
NumReportAbuses	<p>Type int</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description Represents the number of reported abuses on the reply by users. This field is available in API version 24.0 and later.</p>
QuestionId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description ID of the Question to which this reply was made.</p>
UpVotes	<p>Type int</p>
	<p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The total number of up votes for a reply.</p>
VoteTotal	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The total number of all votes for a reply, including up and down votes.</p>

Usage

Use this object to track replies to a Question.

ReplyReportAbuse

Represents a user-reported abuse on a Reply in a Chatter Answers community. This object is available in API version 24.0 and later.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the Reply from which the user reported abuse.</p>
Reason	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The reason the user reported abuse on the Reply, such as <code>Spam</code>, <code>Hateful</code>, or <code>Inappropriate</code>.</p>

Field	Details
ReplyId	Type reference Properties Create, Filter, Group, Sort Description The ID of the Reply from which the user reported abuse.

Usage

Use this object to track user-reported abuse on replies created in a Chatter Answers community.

Report

Represents a report, a set of data that meets certain criteria, displayed in an organized way. Access is read-only. This object is available in API version 20.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `search()`

Fields

Field	Details
Description	Type string Properties Filter, Group, Nillable, Sort Description The description of the report. Limit: 255 characters.
DeveloperName	Type string Properties Filter, Group, Sort Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With

Field	Details
	<p>this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Report Unique Name.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
FolderName	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Name of the folder that contains the report. Available in API version 35.0 and later.</p>
Format	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Required. Indicates the format of the report. This field is available in API version 38.0 and later. Can have one of these values:</p> <ul style="list-style-type: none">• Tabular for reports in that format. The label is <code>Tabular</code>.• Summary for reports in that format. The label is <code>Summary</code>.• Matrix for reports in that format. The label is <code>Matrix</code>.• Multiblock for reports in joined format. The label is <code>Joined</code>.
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>

Field	Details
LastRunDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Returns the date the report was last run. Label is Last Run.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. The report label used in the user interface.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values: <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. This field can't be accessed unless the logged-in user has the "Customize Application" permission. </p>

Field	Details
OwnerId	Type reference Properties Filter Description The ID of the user who currently owns this report.

Supported Query Scopes

Use these scopes to help specify the data your SOQL query returns.

allPrivate

Records saved in all users' private folders.

Requires the user permission "Manage All Private Reports and Dashboards" and [Enhanced Analytics Folder Sharing](#). If your organization was created after the Summer '13 release, you already have Enhanced Analytics Folder Sharing. Available in API version 36.0 and later.

created

Records created by the user running the query.

everything

All records except records saved in other users' private folders.

mine

Records saved in the private folder of the user running the query.

organizationOwned

Records saved in Unfiled Public Reports. In Lightning Experience, the Unfiled Public Reports folder is called Public Reports.

Usage

Use the report object to get report metadata. Query, search, or retrieve specific metadata on reports. Report object fields are read-only.

Example: Reports with "Sales" in Their Name

This SOQL query returns reports that contain the name "Sales" and lists their developer names, format, ID, and report name.

```
SELECT DeveloperName,Format,Id,Name FROM Report WHERE Name LIKE '%Sales%'
```

Example: Reports in an Inactive User's Private Folder

This SOQL query returns reports saved in a specific user's private folder.

```
SELECT Id FROM Report USING SCOPE allPrivate WHERE OwnerId = '005A000000Bc2deFG'
```

SEE ALSO:

[ReportFeed](#)
[ReportTag](#)
[Dashboard](#)

ReportFeed

Represents a single feed item in the feed displayed on a report. This object is available in API version 20.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of ReportFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p>

Field	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by CreatedDateDESC sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>

Field	Details
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p>

Field	Details
	<p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default <code>community</code>, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization. <code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.

Field	Details
	<ul style="list-style-type: none">• AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope.• For feed items with a record parent, users can set NetworkScope only to AllNetworks.• You can't filter a FeedItem on the NetworkScope field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the report record. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none">• ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).

Field	Details
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none">• AdvancedTextPost—created when a user posts a group announcement.• AnnouncementPost—Not used.• ApprovalPost—generated when a user submits an approval.• BasicTemplateFeedItem—Not used.• CanvasPost—a post made by a canvas app posted on a feed.• CollaborationGroupCreated—generated when a user creates a public group.• CollaborationGroupUnarchived—Not used.• ContentPost—a post with an attached file.• CreatedRecordEvent—generated when a user creates a record from the publisher.• DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.• DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RypplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• CaseCommentPost—generated event when a user adds a case comment for a case object• EmailMessageEvent—generated event when an email related to a case object is sent or received• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• ChangeStatusPost—generated event when a user changes the status of a case• AttachArticleEvent—generated event when a user attaches an article to a case <p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p>

Field	Details
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>Visibility can have the following values:</p> <ul style="list-style-type: none"> • AllUsers—The feed item is available to all users who have permission to see the feed item. • InternalUsers—The feed item is available to internal users only. <p>Note the following exceptions for Visibility:</p> <ul style="list-style-type: none"> • For record posts, Visibility is set to InternalUsers for all internal users by default. • External users can set Visibility only to AllUsers. • On user and group posts, only internal users can set Visibility to InternalUsers.

Usage

Use this object to retrieve the current contents of the feed fields, such as type of feed or feed ID.

SEE ALSO:

[Report](#)

ReportTag

Associates a word or short phrase with a Report. This object is available in API version 20.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p>

Field Name	Details
	Properties Create, Filter
	Description ID of the tagged item.
Name	Type string
	Properties Create, Filter
	Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.
TagDefinitionId	Type reference
	Properties Filter
	Description ID of the parent TagDefinition object that owns the tag.
Type	Type picklist
	Properties Create, Filter, Restricted picklist
	Description Defines the visibility of a tag. Valid values:
	<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId.

Usage

ReportTag stores the relationship between its parent TagDefinition and the Report being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

SEE ALSO:

[Report](#)

ReputationLevel

Represents a reputation level defined for a community. This object is available in API version 32.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization. Only users with permissions to create or manage a community can view the ReputationPointsRule records.

Fields

Field Name	Details
Label	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The label for the reputation level.</p>
LevelNumber	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The rank of the reputation level.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the parent community (network) the reputation level applies to.</p>
Threshold	<p>Type double</p> <p>Properties Filter, Sort</p>

Field Name	Details
	Description
	<p>The lower limit of reputation points associated with this reputation level. The maximum number of reputation points a user can accrue is 999,999,999,999,999.</p>

ReputationLevelLocalization

Represents the translated value of a reputation level. Reputation level localization only applies for reputation levels in communities. This object is available in API version 35.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization and reputation is enabled in your community.

Fields

Field Name	Details
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The language the reputation level is translated into. The picklist contains the following fully-supported languages:</p> <ul style="list-style-type: none"> • Chinese (Simplified): zh_CN • Chinese (Traditional): zh_TW • Danish: da • Dutch: nl_NL • English: en_US • Finnish: fi • French: fr • German: de • Italian: it • Japanese: ja • Korean: ko

Field Name	Details
	<ul style="list-style-type: none"> Norwegian: no Portuguese (Brazil): pt_BR Russian: ru Spanish: es Spanish (Mexico): es_MX Swedish: sv Thai: th
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description</p>
	<p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix_componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description</p>
	<p>The ID of the reputation level this translated value applies to.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description</p>
	<p>The translated text for the reputation level. Label is Translation Text.</p>

ReputationPointsRule

Represents the reputation point rules for a community. Each rule specifies an action that community members can earn points from and the points associated with those actions in a particular community. This object is available in API version 32.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

This object is available only if Salesforce Communities is enabled in your organization. Only users with permissions to create or manage a community can view the ReputationPointsRule records.

Fields

Field Name	Details
ParentId	Type reference Properties Filter, Group, Sort Description The ID of the parent community (network) that the point rule applies to.
Points	Type int Properties Filter, Group, Sort Description The reputation points associated with the member action this rule is for. The maximum value this field can contain is 999,999.
Type	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The member action associated with this rule, limited to one of these values: <ul style="list-style-type: none">• Write a post• Write a comment• Receive a comment• Like something

Field Name	Details
	<ul style="list-style-type: none"> • Receive a like • Share a post • Someone shares your post • Mention someone • Receive a mention • Ask a question • Respond to a question • Receive an answer • Mark an answer as best • Your answer is marked as best

ResourceAbsence

Represents a time period in which a service resource is unavailable to work. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service Lightning must be enabled. To create or update resource absences, you need “Read” access to service resources. To delete resource absences, you need “Edit” access to service resources.

Fields

Field Name	Details
AbsenceNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description (Read only) An auto-generated number identifying the absence.</p>
Address	<p>Type address</p> <p>Properties Filter, Nillable</p>

Field Name	Details
	Description
	The compound form of the address associated with the absence.
City	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The city of the address associated with the absence. Maximum length is 40 characters.
Country	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The country of the address associated with the absence. Maximum length is 80 characters.
Description	Type textarea
	Properties
	Create, Nillable, Update
	Description
	The description of the absence.
End	Type dateTime
	Properties
	Create, Filter, Sort, Update
	Description
	The date and time when the absence ends.
GeocodeAccuracy	Type picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.

Field Name	Details
	 Note: This field is available in the API only.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource absence was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource absence was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address associated with the absence. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.  Note: This field is available in the API only.
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address associated with the absence. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.  Note: This field is available in the API only.
Postal Code	Type string

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The postal code of the address associated with the absence. Maximum length is 20 characters.</p>
ResourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The absent service resource.</p>
Start	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The date and time when the absence begins.</p>
State	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state of the address associated with the absence. Maximum length is 80 characters.</p>
Street	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street number and name of the address associated with the absence.</p>
Type	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The type of absence: <i>Meeting</i>, <i>Training</i>, <i>Medical</i>, or <i>Vacation</i>. The default value is <i>Vacation</i>. You can add custom values if needed.</p>

Usage

The Absences related list on a service resource detail page lets you define periods of time when the resource is unavailable to work. Unless you're using the Field Service Lightning managed packages with the scheduling optimizer, service resources can still be assigned appointments that conflict with their absences.

 **Tip:** Create a trigger that sends an approval request to a supervisor when a service resource creates an absence.

If you're not using the Field Service Lightning managed packages, a calendar view isn't available for individual service resources.

ResourceAbsenceFeed

Represents a single feed item on a resource absence record detail page. This object is available in API version 38.0 and later.

A resource absence feed shows changes to tracked fields on a resource absence record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to resource absences in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Resource Absence object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p>

Field Name	Details
	<p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the resource absence record tracked in the feed. The feed is displayed on the detail page for this record.</p>

Field Name	Details
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

ResourceAbsenceHistory

Represents the history of changes made to tracked fields on a resource absence. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ResourceAbsenceId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the resource absence being tracked. The history is displayed on the detail page for this record.</p>

RevenueForecast

Represents a revenue-based forecast.

Use QuantityForecast for quantity-based forecasts.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Closed	<p>Type currency</p> <p>Properties Create, Filter</p> <p>Description Read only. A rollup of opportunities or opportunity line items that have closed in this period.</p>
Commit	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The owner's Commit total.</p>
CommitComment	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description Read only. The comment entered when the owner edited his or her Commit total from the Adjusted Total link on the forecast edit page.</p>
CommitOverride	<p>Type currency</p> <p>Properties Filter, Nillable, Update</p> <p>Description Read only. The owner's override of their own My Commit total.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Filter, Restricted picklist, Update</p>

Field	Details
DefaultRollupCommit	<p>Description Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. You can update or upsert this field and Quota only.</p>
DefaultRollupUpside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description Read only. The owner's standard Commit rollup, including their own opportunities and forecast-level overrides from subordinate users in the role hierarchy.</p>
InvalidationDate	<p>Type dateTime</p> <p>Properties Filter, Nillable</p> <p>Description Read only. If not blank, indicates that the rollup numbers for Forecast Override fields that represent calculated (summarized) amounts may not be up to date.</p>
ManagerChoiceCommit	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist, Update</p> <p>Description Read only. The manager's choice regarding the commit amount:</p> <ul style="list-style-type: none"> • DefaultRollup: Use the manager's default commit rollup for owner's forecast, which reflects the manager's opportunity forecast overrides. • AcceptForecast (default selection): Accept the forecast owner's Adjusted Total commit amount, which may or may not be an override.

Field	Details
	<ul style="list-style-type: none"> ManagerManualOverride: Use the manager's manual override. OpportunityOnlyRollup: Use the opportunity rollup, including opportunity forecast overrides, but excluding any forecast-level (Adjusted Total) overrides.
ManagerChoiceUpside	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist, Update</p> <p>Description Read only. The manager's choice regarding the best case amount:</p> <ul style="list-style-type: none"> DefaultRollup: Use the manager's default best case rollup for owner's forecast, which reflects the manager's opportunity forecast overrides. AcceptForecast (default selection): Accept the forecast owner's Adjusted Total best case amount, which may or may not be an override. ManagerManualOverride: Use the manager's manual override. OpportunityOnlyRollup: Use the opportunity rollup, including opportunity forecast overrides, but excluding any forecast-level (Adjusted Total) overrides.
ManagerClosed	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description Read only. The manager's closed total for the owner's forecast, including any opportunity or opportunity product overrides made by the manager.</p>
ManagerCommit	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The manager's Commit total.</p>
ManagerCommitOverride	<p>Type currency</p> <p>Properties Filter, Nillable, Update</p>

Field	Details
ManagerDefaultRollupCommit	Description Read only. The manager's manual override of the forecast owner's Commit total. Represents an option in the override popup window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerDefaultRollupUpside	Type currency
Properties	Filter, Nillable
ManagerId	Description Read only. The manager's standard Commit rollup for the forecast owner. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerOpportunityRollupCommit	Type currency
Properties	Filter, Nillable
ManagerOpportunityRollupUpside	Description Read only. The manager's standard Best Case rollup for the forecast owner. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerOpportunityRollupUpside	Type reference
Properties	Filter, Nillable
ManagerOpportunityRollupUpside	Description Read only. ID of the direct manager of the user who owns this forecast.
ManagerOpportunityRollupCommit	Type currency
Properties	Filter, Nillable
ManagerOpportunityRollupCommit	Description The manager's view of the forecast owner's opportunity-level Commit rollup, ignoring all forecast overrides. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.
ManagerOpportunityRollupUpside	Type currency

Field	Details
	<p>Properties Filter, Nillable</p> <p>Description The manager's view of the forecast owner's opportunity-level Best Case rollup, ignoring all forecast overrides. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.</p>
ManagerPipeline	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The manager's pipeline total for the owner's forecast, including any opportunity or opportunity product overrides made by the manager.</p>
ManagerTerritoryId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the manager's UserRole or Territory.</p>
ManagerUpside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The manager's Best Case total.</p>
ManagerUpsideOverride	<p>Type currency</p> <p>Properties Filter, Nillable, Update</p> <p>Description The manager's manual override of the forecast owner's Best Case total. Represents an option in the override pop-up window, which allows managers to choose how to roll up the forecast numbers of a direct report.</p>
OpportunityRollupClosed	<p>Type currency</p>

Field	Details
	<p>Properties Filter, Nillable</p> <p>Description The owner's Closed total for his or her opportunities only.</p>
OpportunityRollupCommit	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The owner's Commit total for his or her opportunities only.</p>
OpportunityRollupPipeline	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The owner's Pipeline total for his or her opportunities only.</p>
OpportunityRollupUpside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description Read only. The owner's Best Case total for his or her opportunities only.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter</p> <p>Description ID of the User who owns this forecast. Required on create.</p>
PeriodId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description The ID of the Period that contains the <code>StartDate</code>.</p>
Pipeline	<p>Type currency</p>

Field	Details
	<p>Properties Filter, Nillable</p> <p>Description The total pipeline rollup from subordinates in the role hierarchy, including the owner's opportunities.</p>
ProductFamily	<p>Type picklist</p> <p>Properties Create, Filter, Nillable</p> <p>Description The value chosen in the Product Family picklist, which can be configured from the object management settings for products. This field is relevant if you have chosen Product Families as the Forecast Type in Forecasts Settings. If you are not forecasting by product family or if the forecast represents opportunities that are not associated with a product family, then this field is blank. Otherwise, this field is required on create.</p>
Quota	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The quota amount for the period. You can update or upsert this field and <code>CurrencyIsoCode</code>, and for Quota only, create. Requires the "Modify All Data" and "Manage Users" permission. Required on create.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Nillable</p> <p>Description The start date of this forecast. The period ID of the period that contains this date is written to the <code>PeriodId</code> field if it changes. A new Period is created if none exists. Required on create.</p>
TerritoryId	<p>Type reference</p> <p>Properties Create, Filter, Nillable</p> <p>Description ID of the forecast owner's UserRole or Territory. Required on create if Territory Management is enabled (if this field is available).</p>

Field	Details
Upside	<p>Type currency</p> <p>Properties Filter, Nillable</p> <p>Description The owner's Best Case total.</p>
UpsideComment	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description Read only. The comment entered when the owner edited his or her Best Case total. Label is Best Case Comment.</p>
UpsideOverride	<p>Type currency</p> <p>Properties Filter, Nillable, Update</p> <p>Description Read only. The owner's override of their My Best Case total. Label is Best Case Override.</p>

Usage

Query this object to support customizable forecasts based on revenue. Requires the "View All Data" permission.

You can also update `CurrencyIsoCode` and `Quota`, which means that you can mass update sales users' quotas instead of updating them one by one in the user interface. Editing `Quota` requires "Modify All Data" and "Manage Users" permissions.

The rollup fields always reflect opportunity and opportunity product overrides by the forecast owner or one of the forecast owner's subordinates in the role hierarchy. In addition, the manager rollup fields include overrides by the forecast owner's direct manager in the role hierarchy.

Some of the rollup fields ignore forecast-level (**Adjusted Total**) overrides, but they never ignore opportunity forecast overrides that are visible to the owner or manager.

SEE ALSO:

[QuantityForecast](#)

[RevenueForecastHistory](#)

[Object Basics](#)

RevenueForecastHistory

Represents historical information about revenue-based forecasts that have been submitted (saved) in the user interface.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

- Requires the “View All Data” permission.
- Customer Portal users can’t access this object.

Fields

Field	Details
Closed	Type currency Properties Filter, Nillable Description The closed amount of the forecast.
Commit	Type currency Properties Filter, Nillable Description The commit amount of the forecast.
CommitComments	Type string Properties Filter, Nillable Description Comments about the commit value.
CommitOverridden	Type boolean Properties Defaulted on create, Filter

Field	Details
	Description
	Indicates whether the commit value was overridden (<code>true</code>) or not (<code>false</code>).
CurrencyIsoCode	Type picklist
	Properties
	Defaulted on create, Filter, Restricted picklist
	Description
	Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.
ForecastOverrideId	Type reference
	Properties
	Filter
	Description
	ID of the related forecast override.
Pipeline	Type currency
	Properties
	Filter, Nillable
	Description
	The pipeline amount of the forecast.
Quota	Type currency
	Properties
	Filter, Nillable
	Description
	The quota amount of the forecast.
Upside	Type currency
	Properties
	Filter, Nillable
	Description
	The best case amount of the forecast.
UpsideComments	Type string

Field	Details
	<p>Properties Filter, Nillable</p> <p>Description Comments about the upside value.</p>
UpsideOverridden	<p>Type boolean</p> <p>Properties Defaulted on create Filter</p> <p>Description Indicates whether the upside value was overridden (<code>true</code>) or not (<code>false</code>).</p>

Usage

This is a read-only object specific to customizable forecasting.

When a user submits a revenue-based forecast in the user interface, a new record is created. If the same forecast is ever resubmitted, additional records are added. The `CreatedDate` of a record reflects the day on which the forecast was submitted. This object respects field-level security on the parent object.

SEE ALSO:

[QuantityForecastHistory](#)

[RevenueForecast](#)

[Object Basics](#)

RuleTerritory2Association

Represents a record-assignment rule and its association to an object, such as Account. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your organization's sharing settings. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
IsInherited	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Indicates whether the rule is an <i>inherited</i> rule (<code>true</code>) or a <i>local</i> rule (<code>false</code>). Rule inheritance flows from the parent territory where the rule is created to the rule's descendent territories (if any) in the territory model hierarchy. A local rule is created within a single territory and affects that territory only.</p>
RuleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the rule.</p>
Territory2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the territory where the rule was created.</p>

SamISsoConfig

Represents a SAML Single Sign-On configuration. This object is available in API version 32.0 and later.

Single sign-on is a process that allows network users to access all authorized network resources without having to log in separately to each resource. Single sign-on allows you to validate usernames and passwords against your corporate user database or other client application rather than having separate user passwords managed by Salesforce.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
AttributeFormat	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description For SAML 2.0 only and when <code>identityLocation</code> is set to <code>Attribute</code>. Possible values include <code>unspecified</code>, <code>emailAddress</code> or <code>persistent</code>. All legal values can be found in the “Name Identifier Format Identifiers” section of the Assertions and Protocols SAML 2.0 specification.</p>
AttributeName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the identity provider’s application. Get this name value from your identity provider.</p>
Audience	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The Issuer, also called the “Entity ID.” The value is a URL that uniquely identifies the SAML identity provider.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object’s name in a managed package, and the changes are reflected in a subscriber’s organization.</p>

Field Name	Details
ErrorUrl	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL of the page users should be directed to if there's an error during SAML login. It must be a publicly accessible page, such as a public site Visualforce page. The URL can be absolute or relative.</p>
ExecutionUserID	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user that runs the Apex handler class. The user must have the "Manage Users" permission. A user is required if you specify a SAML JIT handler class.</p>
IdentityLocation	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The location in the assertion where a user is identified. Valid values are:</p> <ul style="list-style-type: none"> • SubjectNameId—The identity is in the <Subject> statement of the assertion. • Attribute—The identity is specified in an <AttributeValue>, located in the <Attribute> of the assertion.
IdentityMapping	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The identifier that the service provider uses for the user during Just-in-Time user provisioning. Valid values are:</p> <ul style="list-style-type: none"> • Username—The user's Salesforce username. • FederationId—The federation ID from the user object; the identifier that's used by the service provider for the user. • UserId—The user ID from the user's Salesforce organization.

Field Name	Details
Issuer	Type string
	Properties Filter, idLookup, Group, Sort
	Description Also called the "Entity ID." The value is a URL that uniquely identifies the SAML identity provider.
Language	Type picklist
	Properties Filter, Group, Restricted picklist, Sort
	Description The language for the organization.
LoginUrl	Type string
	Properties Filter, Nillable, Sort
	Description For SAML 2.0 only: The URL where Salesforce sends a SAML request to start the login sequence.
LogoutUrl	Type string
	Properties Filter, Nillable, Sort
	Description For SAML 2.0 only: The URL to direct users to where they click the Logout link. The default is <code>http://www.salesforce.com</code> .
MasterLabel	Type string
	Properties Filter, Group, Sort
	Description The text that's used to identify the Visualforce page in the Setup area of Salesforce.
NamespacePrefix	Type string

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p>
	<p>The namespace prefix can have one of the following values:</p>
	<ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
OptionsSpInitBinding	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description The service provider initiated request binding, either HTTP Redirect (true) or HTTP POST (false).</p>
OptionsUserProvisioning	<p>Type boolean</p>
	<p>Properties Filter</p>
	<p>Description If true, Just-in-Time user provisioning is enabled, which creates users on the fly the first time that they try to log in. Specify Federation ID for the identityMapping value to use this feature.</p>
RequestSignatureMethod	<p>Type picklist</p>
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description The method that's used to sign the SAML request. Valid values are:</p> <ul style="list-style-type: none"> • RSA-SHA1

Field Name	Details
	<ul style="list-style-type: none"> RSA-SHA256
SamlJitHandlerId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of an existing Apex class that implements the Auth.SamlJitHandler interface.</p>
ValidationCert	<p>Type string</p> <p>Properties Filter, Sort</p> <p>Description The certificate that's used to validate the request. Get this certificate value from your identity provider.</p>
Version	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The SAML version. Valid values are:</p> <ul style="list-style-type: none"> SAML1_1 SAML2_2

Scontrol

A custom s-control, which is custom content that is hosted by the system but executed by the client application.

! **Important:** Visualforce pages supersede s-controls. Organizations that haven't previously used s-controls can't create them. Existing s-controls are unaffected, and can still be edited. We recommend that you move your s-controls to Visualforce. We continue to support the Scontrol object.

Represents a custom s-control, which is custom content that the system hosts, but client applications execute. An s-control can contain any type of content that you can display or run in a Web browser.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update()`

Special Access Rules

- Your organization must be using Enterprise, Developer, or Unlimited Edition and be enabled for custom s-controls.
- Customer Portal users can't access this object.

Fields

Field	Details
Binary	<p>Type base64</p> <p>Properties Nillable, Update</p> <p>Description Binary content of this custom s-control, such as an ActiveX control or a Java archive. Can be specified when created, but not when updated. Limit: 5 MB.</p>
BodyLength	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The length of the custom s-control. Label is Binary Length.</p>
ContentSource	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies the source of the s-control content, either custom HTML, a snippet (s-controls that are included in other s-controls), or a URL.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Description of the custom s-control.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p>

Field	Details
Description	<p>The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is S-Control Name.</p>
 Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.	
EncodingKey	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Picklist of character set encodings, including ISO-08859-1, UTF-8, EUC, JIS, Shift-JIS, Korean (ks_c_5601-1987), Simplified Chinese (GB2312), and Traditional Chinese (Big5).</p>
Filename	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description An uploaded object to display when the custom s-control is added to a custom link. Can be a Java applet, an ActiveX control, or any other type of desired content.</p>
HtmlWrapper	<p>Type textarea</p> <p>Properties Update</p> <p>Description Required. HTML page that will be delivered when the user views this custom s-control. This HTML page can be the entire content of the custom s-control, or it can reference the binary. Limit: 1,048,576 characters. Label is HTML Body.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description Required. Name of this custom s-control. Label is Label.</p>

Field	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
SupportsCaching	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the s-control supports caching (<code>true</code>) or not (<code>false</code>).</p>

Usage

Use custom s-controls to manage custom content that extends application functionality. All users can view custom s-controls, but the “Customize Application” permission is required to create or update custom s-controls.

SEE ALSO:

[Object Basics](#)

ScontrolLocalization

The translated value of the field label for an s-control.

! **Important:** Visualforce pages supersede s-controls. Organizations that haven't previously used s-controls can't create them. Existing s-controls are unaffected, and can still be edited.

When the Translation Workbench is enabled for your organization, provides the translation of the field label of an s-control.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- Your organization must be using Professional, Enterprise, Developer, or Unlimited Edition and be enabled for the Translation Workbench.
- To view this object, you must have the “View Setup and Configuration” permission.

Fields

Field	Details
LanguageLocaleKey	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Restricted picklist</p> <p>Description This field is available in API version 16.0 and earlier. It is the same as the <code>Language</code> field.</p>
Language	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Restricted picklist</p> <p>Description This field is available in API version 17.0 and later. The combined language and locale ISO code, which controls the language for labels displayed in an application. This picklist contains the following fully-supported languages:</p> <ul style="list-style-type: none"> • Chinese (Simplified): <code>zh_CN</code> • Chinese (Traditional): <code>zh_TW</code> • Danish: <code>da</code> • Dutch: <code>nl_NL</code> • English: <code>en_US</code> • Finnish: <code>fi</code> • French: <code>fr</code> • German: <code>de</code> • Italian: <code>it</code>

Field	Details
	<ul style="list-style-type: none">• Japanese: ja• Korean: ko• Norwegian: no• Portuguese (Brazil): pt_BR• Russian: ru• Spanish: es• Spanish (Mexico): es_MX• Swedish: sv• Thai: th
	<p>The following end-user only languages are available.</p> <ul style="list-style-type: none">• Arabic: ar• Bulgarian: bg• Croatian: hr• Czech: cs• English (UK): en_GB• Greek: el• Hebrew: iw• Hungarian: hu• Indonesian: in• Polish: pl• Portuguese (Portugal): pt_PT• Romanian: ro• Slovak: sk• Slovenian: sl• Turkish: tr• Ukrainian: uk• Vietnamese: vi
	<p>The following platform languages are available for organizations that use Salesforce exclusively as a platform.</p> <ul style="list-style-type: none">• Albanian: sq• Arabic (Algeria): ar_DZ• Arabic (Bahrain): ar_BH• Arabic (Egypt): ar_EG• Arabic (Iraq): ar_IQ• Arabic (Jordan): ar_JO• Arabic (Kuwait): ar_KW• Arabic (Lebanon): ar_LB• Arabic (Libya): ar LY

Field	Details
	<ul style="list-style-type: none">• Arabic (Morocco): ar_MA• Arabic (Oman): ar_OM• Arabic (Qatar): ar_QA• Arabic (Saudi Arabia): ar_SA• Arabic (Sudan): ar_SD• Arabic (Syria): ar_SY• Arabic (Tunisia): ar_TN• Arabic (United Arab Emirates): ar_AE• Arabic (Yemen): ar_YE• Armenian: hy• Basque: eu• Bosnian: bs• Bengali: bn• Chinese (Simplified—Singapore): zh_SG• Chinese (Traditional—Hong Kong): zh_HK• English (Australia): en_AU• English (Canada): en_CA• English (Hong Kong): en_HK• English (India): en_IN• English (Ireland): en_IE• English (Malaysia): en_MY• English (Philippines): en_PH• English (Singapore): en_SG• English (South Africa): en_ZA• Estonian: et• French (Belgium): fr_BE• French (Canada): fr_CA• French (Luxembourg): fr_LU• French (Switzerland): fr_CH• Georgian: ka• German (Austria): de_AT• German (Luxembourg): de_LU• German (Switzerland): de_CH• Hindi: hi• Icelandic: is• Irish: ga• Italian (Switzerland): it_CH• Latvian: lv

Field	Details
NamespacePrefix	<ul style="list-style-type: none"> • Lithuanian: lt • Luxembourgish: lb • Macedonian: mk • Malay: ms • Maltese: mt • Romanian (Moldova): ro_MD • Montenegrin: sh_ME • Romansh: rm • Serbian (Cyrillic): sr • Serbian (Latin): sh • Spanish (Argentina): es_AR • Spanish (Bolivia): es_BO • Spanish (Chile): es_CL • Spanish (Colombia): es_CO • Spanish (Costa Rica): es_CR • Spanish (Dominican Republic): es_DO • Spanish (Ecuador): es_EC • Spanish (El Salvador): es_SV • Spanish (Guatemala): es_GT • Spanish (Honduras): es_HN • Spanish (Nicaragua): es_NI • Spanish (Panama): es_PA • Spanish (Paraguay): es_PY • Spanish (Peru): es_PE • Spanish (Puerto Rico): es_PR • Spanish (United States): es_US • Spanish (Uruguay): es_UY • Spanish (Venezuela): es_VE • Tagalog: tl • Tamil: ta • Urdu: ur • Welsh: cy <p>The values in this field are not related to the default locale selection.</p>

Field	Details
	<p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix</i>__<i>componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ScontrolId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Nullable</p> <p>Description</p> <p>The ID of the Scontrol that is being translated.</p>
Value	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Nullable, Update</p> <p>Description</p> <p>The actual translated field label of the s-control. Label is Translation.</p>

Usage

Use this object to translate your s-controls into a supported language. Users with the Translation Workbench enabled can view s-control translations, but either the “Customize Application” or “Manage Translation” permission is required to create or update s-control translations.

SEE ALSO:

[CategoryNodeLocalization](#)
[WebLinkLocalization](#)

SearchPromotionRule

Represents a promoted search term, which is one or more keywords that you associate with a Salesforce Knowledge article. When a user's search query includes these keywords, the associated article is returned first in search results. This object is available in API version 31.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Special Access Rules

A user must have the "Manage Promoted Search Terms" permission.

Fields

Field Name	Details
PromotedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID of the KnowledgeArticleVersion that the promoted search term is associated with. The article must be in published status.</p>
Query	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The text of the promoted search term. Maximum length: 100 characters. You can associate the same promoted search term with multiple articles. If the user's search matches the promoted term, all associated articles are promoted in search results, ordered by relevancy. For best results, create promoted search terms selectively and limit the number of articles that are promoted per term.</p>

Usage

Use this object to optimize article search results in Salesforce Knowledge.

SecureAgent

Represents a Secure Agent that connects Salesforce to on-premises external data sources like SharePoint 2010 and 2013. This object is available in API version 32.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of agent labels in the user interface.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The user-interface name for the agent.</p>

Field Name	Details
Priority	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Specifies the order in which this agent is accessed relative to others in a Secure Agent cluster. This field is available in API version 35 and later.</p>
ProxyUserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user profile specific to the agent.</p>
SecureAgentsClusterId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of a cluster of agents that contains this individual agent. Clusters provide failover protection if an agent on a particular server becomes inaccessible. To edit them, use the SecureAgentsCluster object. This field is available in API version 35 and later.</p>

SecureAgentsCluster

Represents a cluster consisting of several Secure Agents on different servers. Clusters provide failover protection if an agent on a particular server becomes inaccessible. This object is available in API version 35.0 and later.

Secure Agents connect Salesforce to content in on-premises external data sources, such as SharePoint 2010 and 2013. To edit individual agents that are part of a cluster, use the [SecureAgent](#) object.

Supported Calls

`describeLayout(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Describes details about the cluster, providing context to other developers.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language of cluster labels in the user interface.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The user-interface name for the cluster.</p>

SelfServiceUser

Represents a Contact who has been enabled to use your organization's Self-Service portal, where he or she can obtain online support.



Note: Starting with Spring '12, the Self-Service portal isn't available for new orgs. Existing orgs continue to have access to the Self-Service portal.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. All Self-Service users must be associated with a Contact. The contact's email should match the Self-Service user email. The contact must have a value in the <code>AccountId</code> field or an error occurs.</p>
Email	<p>Type email</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Make this the same as the email address for the Contact associated with this SelfServiceUser. Password resets and other system communication will be sent to this email address.</p>
FirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description First name of the Self-Service user.</p>

Field	Details
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the Self-Service user is allowed to log in to the Self-Service portal (<code>true</code>) or not (<code>false</code>). Note that there is no way to delete a Self-Service user. They can only be marked as inactive.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LanguageLocaleKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. This is a restricted picklist field. It is the primary language for the user. All on-screen text in the Self-Service portal is displayed in this language.</p>
LastLoginDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time when the Self-Service user last logged in.</p>
LastName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Last name of the Self-Service user.</p>
LocaleSidKey	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. This is a restricted picklist field. The value of this field affects the formatting and parsing of values, especially numeric values, in the Self-Service portal. Values are two-letter codes that indicate language and sometimes language and country. The codes are based on ISO standards.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Concatenation of FirstName and LastName. Limited to 121 characters.</p>
SuperUser	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this Self-Service user is a super user with additional access on his or her company's Self-Service portal (<code>true</code>) or not (<code>false</code>).</p>
TimeZoneSidKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. This is a restricted picklist field. The time zone of a affects the offset used when displaying or entering times in the Self-Service portal.</p>
Username	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. This contains the name that a Self-Service user enters to log into the Self-Service portal. Value must be unique in your organization. If you try to create or update a user with a duplicate value, the operation is rejected and an error is returned.</p>

Usage

For security reasons, you can't query Self-Service user passwords via the API or the user interface. However, the API allows you to set and reset Self-Service user passwords using the `setPassword()` and `resetPassword()` calls.

SelfServiceUser records created from the API don't cause a notification email to be sent. If you want to notify the user, you must send them an email after creating the user.

SEE ALSO:

[Contact](#)
[User](#)

ServiceAppointment

Represents an appointment to complete field service work for a customer. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`,
`retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The account associated with the appointment. If the parent record is a work order or work order line item, this field's value is inherited from the parent. Otherwise, it remains blank.</p>
ActualDuration	<p>Type int</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The number of minutes that it took the resource to complete the appointment after arriving at the address. When values are first added to the <code>Actual</code></p>

Field Name	Details
	Start and Actual End fields, the Actual Duration is automatically populated to list the difference between the Actual Start and Actual End. If the Actual Start and Actual End fields are subsequently updated, the Actual Duration field doesn't re-update, but you can manually update it.
ActualEndTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The actual date and time the appointment ended.</p>
ActualStartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The actual date and time the appointment started.</p>
Address	<p>Type address</p> <p>Properties Filter</p> <p>Description The address where the appointment is taking place. The address is inherited from the parent record if the parent record lists an address.</p>
AppointmentNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An auto-assigned number that identifies the appointment.</p>
ArrivalWindowEndTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The end of the window of time in which the technician is scheduled to arrive at the site. This window is typically larger than the Scheduled Start and End window</p>

Field Name	Details
	<p>to allow time for delays and scheduling changes. You may choose to share the Arrival Window Start and End with the customer, but keep the Scheduled Start and End internal-only.</p>
ArrivalWindowStartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The beginning of the window of time in which the technician is scheduled to arrive at the site. This window is typically larger than the Scheduled Start and End window to allow time for delays and scheduling changes. You may choose to share the Arrival Window Start and End with the customer, but keep the Scheduled Start and End internal-only.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city where the appointment is completed. Maximum length is 40 characters.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The contact associated with the parent record. If needed, you can manually update the service appointment contact.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country where the work order is completed. Maximum length is 80 characters.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the appointment.</p>

Field Name	Details
DueDate	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The date by which the appointment must be completed. Earliest Start Permitted and Due Date typically reflect terms in the customer's service-level agreement.</p>
Duration	<p>Type double</p> <p>Properties Create, Nillable, Filter, Sort, Update</p> <p>Description The estimated length of the appointment. If the parent record is work order or work order line item, the appointment inherits its parent's duration, but it can be manually updated. The duration is in minutes or hours based on the value selected in the Duration Type field.</p>
DurationType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The unit of the Duration: Minutes or Hours.</p>
EarliestStartTime	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The date after which the appointment must be completed. Earliest Start Permitted and Due Date typically reflect terms in the customer's service-level agreement.</p>
GeocodeAccuracy	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.</p>

Field Name	Details
	<p> Note: This field is available in the API only.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the service appointment was last modified. Its label in the user interface is <code>Last Modified Date</code>.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the service appointment was last viewed.</p>
Latitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>Longitude</code> to specify the precise geolocation of the address where the service appointments is completed. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>
Longitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>Latitude</code> to specify the precise geolocation of the address where the service appointment is completed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>
OwnerId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>
	<p>Description The owner of the service appointment.</p>
ParentRecordId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description The parent record associated with the appointment.</p>
ParentRecordStatusCategory	<p>Type string</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description (Read only) The status Category of the parent record. If the parent record is a work order or work order line item, this field is populated; otherwise, it remains blank.</p>
ParentRecordType	<p>Type string</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description (Read only) The type of parent record: Account, Asset, Opportunity, Work Order, or Work Order Line Item.</p>
PostalCode	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The postal code where the work order is completed. Maximum length is 20 characters.</p>
SchedEndTime	<p>Type dateTime</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>

Field Name	Details
SchedStartTime	<p>Description The time at which the appointment is scheduled to end. If you are using the Field Service Lightning managed packages with the scheduling optimizer, this field is populated once the appointment is assigned to a resource. <code>Scheduled_End - Scheduled_Start = Estimated Duration</code>.</p>
ServiceTerritoryId	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The time at which the appointment is scheduled to start. If you are using the Field Service Lightning managed packages with the scheduling optimizer, this field is populated once the appointment is assigned to a resource.</p>
State	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The service territory associated with the appointment. If the parent record is a work order or work order line item, the appointment inherits its parent's service territory.</p>
Status	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state where the service appointment is completed. Maximum length is 80 characters.</p>

Field Name	Details
	<ul style="list-style-type: none"> • Dispatched—Assigned service resource has been notified about their assignment. • In Progress—Work has begun. • Completed—Work is complete. • Cannot Complete—Work could not be completed. • Canceled—Work is canceled, typically before any work began
StatusCategory	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The category that each <code>Status</code> value falls into. The <code>Status Category</code> field has seven values which are identical to the default <code>Status</code> values. If you create custom <code>Status</code> values, you must indicate which category it belongs to. For example, if you create a <code>Customer Absent</code> value, you may decide that it belongs in the <code>Cannot Complete</code> category. The <code>Status Category</code> field can be useful to reference in custom apps, triggers, and validation rules. Status categories let you extend and customize the work life cycle while still maintaining a consistent work classification for tracking, reporting, and business process management.</p>
Street	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street number and name where the service appointment is completed.</p>
Subject	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A short phrase describing the appointment.</p>

Usage

Service appointments always have a parent record, which can be a work order, work order line item, opportunity, account, or asset. The type of parent record tells you about the nature of the service appointment:

- Service appointments on *work orders* and *work order line items* offer a more detailed view of the work being performed. While work orders and work order line items let you enter general information about a task, service appointments are where you add the details about scheduling and ownership.
- Service appointments on *assets* represent work being performed on the asset.
- Service appointments on *accounts* represent work being performed for the account.
- Service appointments on *opportunities* represent work that is related to the opportunity.

ServiceAppointmentFeed

Represents a single feed item on a service appointment record detail page. This object is available in API version 38.0 and later.

A service appointment feed shows changes to tracked fields on a service appointment record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service appointments in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Operating Hours object
- "Moderate Chatter"

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>

Field Name	Details
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service appointment record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>

Field Name	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

ServiceAppointmentHistory

Represents the history of changes made to tracked fields on a service appointment. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p>

Field Name	Details
	<p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ServiceAppointmentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service appointment being tracked. The history is displayed on the detail page for this record.</p>

ServiceAppointmentOwnerSharingRule

Represents the rules for sharing a service appointment with user records other than the owner or anyone above the owner in the role hierarchy. This object is available in API version 38.0 and later.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. A service appointment owned by a User in the source Group triggers the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
ServiceAppointmentAccessLevel	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of access granted to the target Group, or UserRole. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the User or Group being granted access.</p>

ServiceAppointmentShare

Represents a sharing entry on a service appointment. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the user or group has to the service appointment. The possible values are:</p> <ul style="list-style-type: none"> • Read

Field Name	Details
	<ul style="list-style-type: none"> • <i>Edit</i> • <i>All</i> (This value isn't valid for create or update calls.)
	<p>Set to an access level that is at least equal to the organization's default service appointment access level.</p>
ParentId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description The service appointment associated with the sharing entry.</p>
RowCause	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description The reason why this sharing entry exists. You can write to this field only when its value is omitted or set to <i>Manual</i> (default). Valid values include:</p> <ul style="list-style-type: none"> • <i>Manual</i>—The user or group has access because a user with "All" access manually shared the service appointment. • <i>Owner</i>—The user is the owner of the service appointment. • <i>Team</i>—The user or group has team access. • <i>Rule</i>—The user or group has access via a service appointment sharing rule.
UserOrGroupId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description (Read only) ID of the user or group that has access to the service appointment.</p>

ServiceChannel

Represents a channel of work items that are received from your organization—for example, cases, chats, or leads. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CapacityPercentage	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel.</p> <p>For example, you might give phone calls a capacity percentage of <code>100</code>. If an agent receives a phone call, the agent won't receive new work items until the call ends, because at that point the agent's capacity will have reached 100%.</p> <p>This field is available in API version 32.0 and earlier. For later API versions, you can set the capacity percentage of work items on the QueueRoutingConfig object.</p>
CapacityWeight	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel.</p> <p>For example, if an agent has a capacity of <code>6</code>, and cases are assigned a capacity weight of <code>2</code>, an agent can be assigned up to 3 cases before the agent is at capacity and can't receive new work items.</p> <p>This field is available in API version 32.0 and earlier. For later API versions, you can set the capacity weight of work items on the QueueRoutingConfig object.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>

Field	Details
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the service channel.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the service channel.</p>
RelatedEntity	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Unique, Update</p> <p>Description The type of object that's associated with this service channel.</p>

ServiceChannelStatus

Represents the status that's associated with a specific service channel. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), query(), retrieve(), update()`

Fields

Field	Details
ServiceChannelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the service channel.</p>

Field	Details
ServicePresenceStatusId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the presence status that's associated with the service channel that's specified by the ServicePresenceChannelId.</p>

ServiceContract

Represents a customer support contract (business agreement). This object is available in API version 18.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Field	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the account associated with the service contract.</p>
ActivationDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The initial day the service contract went into effect (whereas startDate may include a renewal date).</p>
ApprovalStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Sort</p>

Field	Details
	Description
	Approval status of the service contract.
BillingAddress (beta)	Type
	address
	Properties
	Filter, Nillable
	Description
	The compound form of the billing address. Read-only. See Address Compound Fields for details on compound address fields.
BillingCity	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address. Maximum size is 40 characters.
BillingCountry	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	Details for the billing address. Maximum size is 40 characters.
BillingCountryCode	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The ISO country code for the service contract's billing address.
BillingLatitude	Type
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.

Field	Details
BillingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
BillingPostalCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details for the billing address. Maximum size is 20 characters.</p>
BillingState	<p>Type string</p> <p>Properties Group, Sort, Filter, Nillable</p> <p>Description Details for the billing address. Maximum size is 20 characters.</p>
BillingStateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code for the service contract's billing address.</p>
BillingStreet	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Street address for the billing address.</p>
ContactId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description Required. ID of the Contact associated with the service contract. Must be a valid ID.</p>
ContractNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description Unique number automatically assigned to the service contract.</p>
Description	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description Description of the service contract.</p>
Discount	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The last day the service contract is in effect. Weighted average of all contract line item discounts on the service contract.</p>
EndDate	<p>Type date</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The last day the service contract is in effect.</p>
GrandTotal	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Total price of the service contract plus shipping and taxes.</p>
IsDeleted	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
LineItemCount	<p>Type int</p> <p>Properties Filter, Nillable, Group, Sort</p> <p>Description Number of ContractLineItem records associated with the service contract.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name of the service contract.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The ID of the user who currently owns the service contract.</p>

Field	Details
ParentServiceContractId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The service contract's parent service contract, if it has one.</p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description ID of the Pricebook2 associated with the service contract. Must be a valid ID.</p>
RootServiceContractId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The top-level service contract in a service contract hierarchy. Depending on where a service contract lies in the hierarchy, its root could be the same as its parent.</p>
ShippingAddress (beta)	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the shipping address. Read-only. See Address Compound Fields for details on compound address fields.</p>
ShippingCity	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Details of the shipping address. Maximum size is 40 characters.</p>
ShippingCountry	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description Details of the shipping address. Country maximum size is 40 characters.</p>
ShippingCountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code for the service contract's shipping address.</p>
ShippingLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p>
ShippingLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p>
ShippingPostalCode	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Details of the shipping address. Postal code maximum size is 20 characters.</p>
ShippingState	<p>Type string</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description Details of the shipping address. State maximum size is 20 characters.</p>

Field	Details
ShippingStateCode	Type picklist
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ISO state code for the service contract's shipping address.
ShippingStreet	Type textarea
	Properties Create, Filter, Nillable, Update
	Description The street address of the shipping address. Maximum of 255 characters.
SpecialTerms	Type textarea
	Properties Create, Nillable, Update
	Description Any terms specifically agreed to and tracked in the service contract.
StartDate	Type date
	Properties Create, Filter, Nillable, Update
	Description The first day the service contract is in effect.
Status	Type picklist
	Properties Filter, Nillable
	Description The status of the service contract, such as Inactive.
Subtotal	Type currency
	Properties Filter, Nillable

Field	Details
	Description Total of the service contract line items (products) before discounts, taxes, and shipping are applied.
Tax	Type currency Properties Create, Filter, Nillable, Update Description Total taxes for the service contract.
Term	Type int Properties Create, Filter, Nillable, Update Description Number of months that the service contract is valid.
TotalPrice	Type currency Properties Filter, Nillable Description Total of the contract line items (products) after discounts and before taxes and shipping.

Usage

Use this object to query and manage service contracts.

SEE ALSO:

- [ServiceContractHistory](#)
- [ServiceContractShare](#)
- [ServiceContractOwnerSharingRule](#)

ServiceContractFeed

Represents a single feed item in the feed displayed on the detail page for a service contract record. This object is available in API version 23.0 and later.

A service contract feed shows recent changes to a service contract for any fields that are tracked in feeds, and comments and posts about the record. It's a useful way to stay up-to-date with changes to service contracts.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the ServiceContract object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content for the ServiceContractFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p>Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>

Field Name	Details
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if <code>Type</code> is <code>ContentPost</code>. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	Description
	Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.
ContentType	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the user who added this object to the feed. For example, if a client application migrates multiple posts and comments from another application into a feed, <code>InsertedById</code> is set to the ID of the logged-in user.
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.
	Rich text supports the following HTML tags:
	<ul style="list-style-type: none"> • <p>
	 Tip: Though the tag isn't supported, you can use <p> </p> to create lines.
	<ul style="list-style-type: none"> •
	<ul style="list-style-type: none"> • <code>
	<ul style="list-style-type: none"> • <i>
	<ul style="list-style-type: none"> • <u>
	<ul style="list-style-type: none"> • <s>
	<ul style="list-style-type: none"> •
	<ul style="list-style-type: none"> •

Field Name	Details
	<ul style="list-style-type: none"> • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description This field is a standard system field. When a feed item is created, LastModifiedDate is the same as CreatedDate. If a FeedComment is inserted on the feed item, then LastModifiedDate becomes the CreatedDate for the FeedComment. Deleting the FeedComment does not change the LastModifiedDate. Ordering by LastModifiedDate DESC sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the <code>FeedItem</code> is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a <code>FeedItem</code> on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the service contract record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the <code>ContentVersion</code> object associated with a <code>ContentPost</code>. This field is null for all posts except <code>ContentPost</code>. For example, set this field to an existing <code>ContentVersion</code> and post it to a feed as a <code>ServiceContractFeed</code> object of <code>Type ContentPost</code>.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>Title of the <code>ServiceContractFeed</code>. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL and this field is the link name.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of ServiceContractFeed:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field.

Field Name	Details
	<ul style="list-style-type: none"> • <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none"> • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case. • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case. • <code>ChatTranscriptPost</code>—generated event when Live Agent transcript is saved to a case. • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case.
Visibility	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • <code>Visibility</code> can be updated on record posts. • The <code>update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for a service contract record.

SEE ALSO:

[ServiceContract](#)
[ServiceContractHistory](#)
[ServiceContractOwnerSharingRule](#)
[FeedItem](#)

ServiceContractHistory

Represents the history of changes to the values in the fields on a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The name of the field that was changed.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable</p>

Field	Details
	Description The new value of the field that was changed.
OldValue	Type anyType Properties Nillable Description The latest value of the field before it was changed.
ServiceContractId	Type reference Properties Filter Description Required. ID of the ServiceContract.

Usage

Use this object to identify changes to a service contract.

This object respects field level security on the parent object.

SEE ALSO:

[ServiceContract](#)
[ServiceContractShare](#)
[ServiceContractOwnerSharingRule](#)

ServiceContractOwnerSharingRule

Represents the rules for sharing a ServiceContract (customer service agreement) with users other than the owner. This object is available in API version 18.0 and later.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description A value that represents the type of sharing allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort,</p>

Field Name	Details
	<p>Description The ID representing the source group. Service contracts owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the user interface.</p>
UserorGroupId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description The ID representing the target user or group. Target users or groups are given access.</p>

Usage

Use this object to manage the sharing rules for a service contract. General sharing and territory management-related sharing use this object.

SEE ALSO:

[ServiceContract](#)
[ServiceContractShare](#)
[ServiceContractHistory](#)
[Metadata API Developer Guide: SharingRules](#)

ServiceContractShare

Represents a sharing entry on a ServiceContract (customer support agreement). This object is available in API version 18.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Level of access that the User or Group has to the ServiceContract. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All (This value is not valid for create or update calls.) <p>This field must be set to an access level that is higher than the organization's default access level for service contracts.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the parent object, if any.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Defaulted on create, Restricted picklist</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Values may include:</p> <ul style="list-style-type: none"> • Manual—The User or Group has access because a user with "All" access manually shared the ServiceContract with them. • Owner—The User is the owner of the ServiceContract or is in a role above the service contract owner in the role hierarchy.
UserOrGroupId	<p>Type reference</p>

Field	Details
Properties	Create, Filter
Description	ID of the User or Group that has been given access to the ServiceContract.

Usage

This object allows you to determine which users and groups can view and edit ServiceContract records owned by other users.

If you attempt to create a new record that matches an existing record, the create call updates any modified fields and returns the existing record.

SEE ALSO:

[ServiceContract](#)
[ServiceContractHistory](#)
[ServiceContractOwnerSharingRule](#)

ServicePresenceStatus

Represents a presence status that can be assigned to a service channel. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the presence status.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the presence status.</p>

ServiceResource

Represents a service technician or dispatcher in field service. This object is available in API version 38.0 and later.

Supported Calls

`create()`,`describeLayout()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`search()`,`update()`,`upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the resource.</p>

Field Name	Details
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description When selected, this option means that the resource can be assigned to work orders. For service tracking purposes, resources can't be deleted, so deactivating a resource is the best way to send them into retirement.</p>
IsCapacityBased	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Capacity-based resources are limited to a certain number of hours or appointments in a specified time period.</p>
	<p> Tip: The Capacities related list shows a resource's capacity.</p>
IsOptimizationCapable	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description When selected, this option means that the service scheduling optimizer can assign this resource to work orders during the optimization process. Use only if the Field Service Lightning managed packages are installed. Its label in the user interface is Include in Scheduling Optimization. Only users with the "Field Service Scheduling" permission set license can be included in scheduling optimization.</p>
LastKnownLatitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The latitude of the last known location.</p>
LastKnownLongitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field Name	Details
	Description
	The longitude of the last known location.
LastKnownLocation	Type location
	Properties
	Nillable
	Description
	The service resource's last known location. You can configure this field to display data collected from a custom mobile app. This field is not visible in the user interface, but you can expose it on service resource page layouts or set up field tracking to be able to view a resource's location history.
LastKnownLocationDate	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date and time of the last known location.
LastReferencedDate	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date when the service resource was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date when the service resource was last viewed.
Name	Type string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	The resource's name. You'll likely want this to be the name or title of the associated user.

Field Name	Details
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The owner of the service resource.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The associated user.</p>
ResourceType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates whether a resource is a technician or a dispatcher. Resources who are dispatchers can't be capacity-based or included in scheduling optimization. Only users with the "Field Service Dispatcher" permission set license can be dispatchers.</p> <p> Note: You can't add additional resource types.</p>

Usage

View service resources on the Service Resources tab. In addition:

- Service resources that are assigned to a service appointment appear in the Assigned Resources related list on the appointment detail page. You can assign multiple service resources to an appointment.
- Service resources that belong to a service territory appear in the Service Territory Members related list on the territory detail page.

For tracking purposes, service resources can only be deactivated, not deleted. To deactivate a service resource, deselect **Active** on their detail page.

Deactivating a user deactivates the related service resource. You can't create a service resource that is linked to an inactive user.

ServiceResourceCapacity

Represents the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeLayout(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Fields

Field Name	Details
CapacityInHours	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The number of hours that the resource can work per time period.</p>
CapacityInWorkItems	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p>
CapacityNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description (Read only) An auto-generated number identifying the capacity record.</p>
EndDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The date the capacity ends; for example, the end date of a contract.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>

Field Name	Details
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
ServiceResourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The associated service resource.</p>
StartDate	<p>Type date</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The date the capacity goes into effect.</p>
TimePeriod	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Days, Hours, or Months. For example, if a resource can work 80 hours per month, the capacity's Time Period would be <code>Month</code> and Hours per Time Period would be <code>80</code>.</p>

Usage

Service resources who are capacity-based can only work a certain number of hours or complete a certain number of service appointments within a specified time period. Contractors tend to be capacity-based. To indicate that a service resource is capacity-based, select **Capacity-Based** on the service resource detail page, then define their capacity in the Capacities related list:

1. Click **New Resource Capacity**.
2. Enter a start date and an end date to indicate when the capacity is in effect for the resource. For example, if the capacity represents a six-month contract, enter the contract's start and end dates.
3. Specify how much the resource can work:

- Select the `Time Period` that the capacity is based on: hours, days, or months. For example, if the resource can work 80 hours per month, select **Month**.
- If you want the resource's capacity to be based on the number of hours worked, fill out `Hours per Time Period`. For example, if the resource can work 80 hours per month, enter `80`.
- If you want the resource's capacity to be based on the number of service appointments they are assigned to, fill out `Work Items per Time Period`. For example, if the resource can complete 20 appointments per month, enter `20`.



Note: If you're using the Field Service Lightning managed packages and would like to measure capacity both in hours and in number of work items, enter a value for both. The resource is considered to reach their capacity based on whichever term is met first—hours or number of work items.

4. Click **Save**. You can set multiple capacities for a resource as long as their start and end dates do not overlap.



Important: If you aren't using the Field Service Lightning managed packages, capacity serves more as a suggestion than a rule. Resources can still be scheduled beyond their capacity, and you aren't notified when a resource exceeds their capacity.

ServiceResourceCapacityFeed

Represents a single feed item on a service resource capacity record detail page. This object is available in API version 38.0 and later.

A service resource capacity feed shows changes to tracked fields on a service resource capacity record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service resource capacities in Salesforce.

Supported Calls

`delete()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Operating Hours object
- "Moderate Chatter"

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>

Field Name	Details
CommentCount	Type int
	Properties Filter, Group, Sort
	Description The number of comments associated with this feed item.
InsertedById	Type reference
	Properties Group, Nullable, Sort
	Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.
IsRichText	Type boolean
	Properties Defaulted on create, Filter, Group, Sort
	Description true indicates that the body of the feed contains rich text. false indicates that the post is rendered as plain text.
LikeCount	Type int
	Properties Filter, Group, Sort
	Description The number of feed likes associated with this feed item.
LinkUrl	Type url
	Properties Nullable, Sort
	Description The URL of a link post.
ParentId	Type reference
	Properties Filter, Group, Sort

Field Name	Details
	Description
	ID of the service resource capacity record tracked in the feed. The feed is displayed on the detail page for this record.
RelatedRecordId	Type reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.
Title	Type string
	Properties
	Group, Nillable, Sort
	Description
	The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.
Type	Type picklist
	Properties
	Filter, Group, Nillable, Restricted picklist, Sort
	Description
	The type of feed item.

ServiceResourceCapacityHistory

Represents the history of changes made to tracked fields on a service resource capacity record. This object is available in API version 38.0 and later.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and field tracking for service resource capacity fields must be configured.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ServiceResourceCapacityId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service resource capacity being tracked. The history is displayed on the detail page for this record.</p>

ServiceResourceFeed

Represents a single feed item on a service resource record detail page. This object is available in API version 38.0 and later.

A service resource feed shows changes to tracked fields on a service resource record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service resources in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Operating Hours object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>

Field Name	Details
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service resource record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	Description
	<p>The type of feed item.</p>

ServiceResourceHistory

Represents the history of changes made to tracked fields on a service resource. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>

Field Name	Details
ServiceResourceId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service resource being tracked. The history is displayed on the detail page for this record.</p>

ServiceResourceOwnerSharingRule

Represents the rules for sharing a service resource with user records other than the owner or anyone above the owner in the role hierarchy. This object is available in API version 38.0 and later.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With</p>

Field	Details
	<p>this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p>
	<p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description The ID representing the source group. A service resource owned by a User in the source Group triggers the rule to give access.</p>
Name	<p>Type string</p>
	<p>Properties Create, Filter, Group, Sort, Update</p>
	<p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
ServiceResourceAccessLevel	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
	<p>Description A value that represents the type of access granted to the target Group, or UserRole. The possible values are:</p>
	<ul style="list-style-type: none"> • Read • Edit • All
UserOrGroupId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description The ID representing the User or Group being granted access.</p>

ServiceResourceShare

Represents a sharing entry on a service resource. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the user or group has to the service resource. The possible values are:</p> <ul style="list-style-type: none">• <i>Read</i>• <i>Edit</i>• <i>All</i> (This value isn't valid for create or update calls.) <p>Set to an access level that is at least equal to the organization's default service resource access level.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The service resource associated with the sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description The reason why this sharing entry exists. You can write to this field only when its value is omitted or set to <i>Manual</i> (default). Valid values include:</p>

Field Name	Details
	<ul style="list-style-type: none"> • <i>Manual</i>—The user or group has access because a user with “All” access manually shared the service resource. • <i>Owner</i>—The user is the owner of the service resource. • <i>Team</i>—The user or group has team access. • <i>Rule</i>—The user or group has access via a service resource sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description (Read only) ID of the user or group that has access to the service resource.</p>

ServiceResourceSkill

Represents a skill that a service resource possesses. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
EffectiveEndDate	<p>Type date</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description The date when the skill expires. For example, if a service resource needs to be re-certified after six months, the end date would be the date their certification expires.</p>
EffectiveStartDate	<p>Type date</p>

Field Name	Details
	Properties
	Create, Filter, Sort, Update
	Description
	The date when the service resource gains the skill. For example, if the skill represents a certification, the start date would be the date of certification.
LastReferencedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date when the resource skill was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date when the resource skill was last viewed.
ServiceResourceId	Type
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	The service resource who possesses the skill.
SkillId	Type
	reference
	Properties
	Create, Filter, Group, Sort, Update
	Description
	The skill the service resource possesses.
SkillLevel	Type
	double
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The service resource's skill level. Skill level can range from zero to 99.99.

Field Name	Details
SkillNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An auto-generated number identifying the resource skill assignment.</p>

Usage

You can assign skills to all service resources in your org to indicate their certifications and areas of expertise, and specify each resource's skill level from 0 to 99.99. For example, you can assign Maria the "Welding" skill, level 50.

If you intend to use the skills feature, determine which skills you want to track and how skill level should be determined. For example, you may want the skill level to reflect years of experience, certification levels, or license classes.

ServiceResourceSkillFeed

Represents a single feed item on a service resource skill record detail page. This object is available in API version 38.0 and later.

A service resource skill feed shows changes to tracked fields on a service resource skill record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service resource skills in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Operating Hours object
- "Moderate Chatter"

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	Description
	<p>The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	Type
	int
	Properties
	Filter, Group, Sort
	Description
	The number of comments associated with this feed item.
InsertedById	Type
	reference
	Properties
	Group, Nullable, Sort
	Description
	ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	true indicates that the body of the feed contains rich text. false indicates that the post is rendered as plain text.
LikeCount	Type
	int
	Properties
	Filter, Group, Sort
	Description
	The number of feed likes associated with this feed item.
LinkUrl	Type
	url
	Properties
	Nullable, Sort
	Description
	The URL of a link post.

Field Name	Details
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service resource skill record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

ServiceResourceSkillHistory

Represents the history of changes made to tracked fields on a service resource skill. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ServiceResourceSkillId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the resource skill being tracked. The history is displayed on the detail page for this record.</p>

ServiceTerritory

Represents a region in which field service work can be performed. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Address	<p>Type address</p> <p>Properties Filter</p> <p>Description An address to associate with the territory. You may want to list the address of the territory's headquarters.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city of the associated address. Maximum length is 40 characters.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country to associate with the territory. Maximum length is 80 characters.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the territory.</p>
GeocodeAccuracy	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.</p> <p> Note: This field is available in the API only.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the service territory is meant to be used. If a territory is inactive, you can't add members to it or link it to work orders, work order line items, or service appointments.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the territory was last modified. Its label in the user interface is Last Modified Date.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the territory was last viewed.</p>
Latitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with Longitude to specify the precise geolocation of the address associated with the territory. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>

Field Name	Details
Longitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with <code>Latitude</code> to specify the precise geolocation of the address associated with the territory. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the territory.</p>
OperatingHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The territory's operating hours, which indicate when service appointments within the territory can occur. Service resources who are members of a territory automatically inherit the territory's operating hours unless different hours are specified on the resource record.</p>
ParentTerritoryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The territory's parent service territory, if it has one. For example, a <i>Northern California</i> territory can have a <i>State of California</i> territory as its parent. A service territory hierarchy can contain up to 500 territories.</p>
PostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	Description
	The postal code of the address associated with the territory. Maximum length is 20 characters.
State	Type string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The state of the address associated with the territory. Maximum length is 80 characters.
Street	Type textarea
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The street number and name of the address associated with the territory.
TopLevelTerritoryId	Type reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	(Read only) The top-level territory in a hierarchy of service territories. Depending on where a territory lies in the hierarchy, its top-level territory might be the same as its parent.

Usage

View service territories on the Service Territories tab. You can also view a service resource's territories on the Service Territories related list on the resource detail page.

If you want to use service territories, determine which territories you need to create. Depending on how your business works, you may decide to create territories based on cities, counties, or other factors. If you plan to build out a hierarchy of service territories, create the highest-level territories first.

Create service territories from the Service Territories tab in Salesforce. After you create a territory, you can add members to it via the Service Territory Members related list. Service territory members are resources who work within the territory, and associating them with a territory ensures that they're assigned to appointments near their home base.

For example, you can create a hierarchy of territories to represent the areas where your team works in California. Include a top-level territory named *California*, three child territories named *Northern California*, *Central California*, and *Southern California*, and a series of third-level territories corresponding to California counties. Assign service resources to each county territory to indicate who is available to work in that county.

ServiceTerritoryFeed

Represents a single feed item on a service territory record detail page. This object is available in API version 38.0 and later.

A service territory feed shows changes to tracked fields on a service territory record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service territories in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Operating Hours object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	Description
	ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	true indicates that the body of the feed contains rich text. false indicates that the post is rendered as plain text.
LikeCount	Type
	int
	Properties
	Filter, Group, Sort
	Description
	The number of feed likes associated with this feed item.
LinkUrl	Type
	url
	Properties
	Nillable, Sort
	Description
	The URL of a link post.
ParentId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	ID of the service territory record tracked in the feed. The feed is displayed on the detail page for this record.
RelatedRecordId	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.

Field Name	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

ServiceTerritoryHistory

Represents the history of changes made to tracked fields on a service territory. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p>

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ServiceTerritoryId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service territory being tracked. The history is displayed on the detail page for this record.</p>

ServiceTerritoryMember

Represents a service resource who is able to be assigned to service appointments in the service territory. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Address	<p>Type address</p> <p>Properties Filter</p>

Field Name	Details
	Description
	The member's address. You may want to list the related service resource's address in this field.
City	Type string
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Sort, Update
	Description
	The city of the member's address. Maximum length is 40 characters.
Country	Type string
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Sort, Update
	Description
	The country of the member's address. Maximum length is 80 characters.
EffectiveEndDate	Type date
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The date when the service resource is no longer a member of the territory. If the resource will be working in the territory for the foreseeable future, leave this field blank. This field is mainly useful for indicating when a temporary relocation ends.
EffectiveStartDate	Type date
	Properties
	Create, Filter, Sort, Update
	Description
	The date when the service resource becomes a member of the service territory.
GeocodeAccuracy	Type picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.

Field Name	Details
	 Note: This field is available in the API only.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the territory member was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the territory member was last viewed.
Latitude	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the member's address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.  Note: This field is available in the API only.
Longitude	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the member's address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.  Note: This field is available in the API only.
MemberNumber	Type string

Field Name	Details
	<p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description (Read only) An auto-generated number identifying the service territory member.</p>
OperatingHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Nillable, Update</p> <p>Description The operating hours assigned to the service territory member. If no operating hours are specified, the member is assumed to use their parent service territory's operating hours. If a member needs special operating hours, create them in Setup and select them in the <code>Operating Hours</code> lookup field on the member's detail page.</p>
PostalCode	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description The postal code of the member's address. Maximum length is 20 characters.</p>
ServiceResourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The service resource assigned to the service territory.</p>
ServiceTerritoryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The service territory that the service resource is assigned to.</p>
State	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	Description The state of the member's address. Maximum length is 80 characters.
Street	Type textarea
	Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update
	Description The street number and name of the member's address.
TerritoryType	Type picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Primary, Secondary, or Relocation.
	<ul style="list-style-type: none"> • The primary territory is typically the territory where the resource works most often—for example, near their home base. Service resources can only have one primary territory. • Secondary territories are territories where the resource can be assigned to appointments if needed. Service resources can have multiple secondary territories. • Relocation territories represent temporary moves for service resources. If you're using the Field Service Lightning managed packages with the scheduling optimizer, resources with relocation territories are always assigned to services within their relocation territories during the specified relocation dates; if they don't have a relocation territory, the primary territories are favored over the secondary.
	For example, a service resource might have the following territories:
	<ul style="list-style-type: none"> • Primary territory: <i>West Chicago</i> • Secondary territories: <ul style="list-style-type: none"> — <i>East Chicago</i> — <i>South Chicago</i> • Relocation territory: <i>Manhattan</i>, for a three-month period

Usage

If you delete a service territory with members, the service resources who were members no longer have any connection to the territory.

ServiceTerritoryMemberFeed

Represents a single feed item on a service territory member record detail page. This object is available in API version 38.0 and later.

A service territory member feed shows changes to tracked fields on a service territory member record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to service territory members in Salesforce.

Supported Calls

`delete()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Operating Hours object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p>

Field Name	Details
	Description
	<p>ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	Type
	boolean
	Properties
	Defaulted on create, Filter, Group, Sort
	Description
	<p><code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	Type
	int
	Properties
	Filter, Group, Sort
	Description
	The number of feed likes associated with this feed item.
LinkUrl	Type
	url
	Properties
	Nillable, Sort
	Description
	The URL of a link post.
ParentId	Type
	reference
	Properties
	Filter, Group, Sort
	Description
	<p>ID of the service territory member record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	Type
	reference
	Properties
	Group, Nillable, Sort
	Description
	<p>ID of the <code>ContentVersion</code> object associated with a content post. This field is null for all posts, except <code>ContentPost</code>.</p>

Field Name	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

ServiceTerritoryMemberHistory

Represents the history of changes made to tracked fields on a service territory member. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>

Field Name	Details
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
ServiceTerritoryMemberId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the service territory member being tracked. The history is displayed on the detail page for this record.</p>

SessionPermSetActivation

The SessionPermSetActivation object represents a permission set assignment activated during an individual user session. When a SessionPermSetActivation object is inserted into a permission set, an activation event fires, allowing the permission settings to apply to the user's specific session. This object is available in API versions 37.0 and later as part of the Session-Based Permission Sets Developer Preview.

 **Note:** Session-Based Permission Sets is currently available as a Developer Preview.

 **Important:** Managed packages with permission sets that require session activation cannot be installed on customer orgs without this feature.

Supported Calls

`describeCompactLayouts(), describeLayout(), query(), retrieve()`

Fields

Field Name	Details
AuthSessionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The session ID related to this permission set assignment for its duration.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The session details, such as device used and browser.</p>
PermissionSetId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The permission set ID related to this permission set assignment and user for its duration.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The user ID of the user to whom this permission set assignment applies for its duration.</p>

Usage

Use SessionPermSetActivation to create a permission set available only for a specified session's duration. For example, create permission sets that provide access to specific applications only during authenticated sessions.

In the following Apex example, an identified session is activated after session information is submitted via a button. Successful activation results in a confirmation message displayed to the user.

```
public class SessionPermSetActivationController {
    // id of the session permission set to be activated
```

```

private final String sessionPermSetId = '0PSxx00000004rJ';
private final String sessionId;

public SessionPermSetActivationController() {
    Map<String, String> sessionManagement = Auth.SessionManagement.getCurrentSession();

    sessionId = sessionManagement.get('SessionId');
}

public PageReference activate() {
    // activate the permission set
    SessionPermSetActivation activation = new SessionPermSetActivation();
    activation.AuthSessionId = sessionId;
    activation.PermissionSetId = sessionPermSetId;
    activation.Description = 'created by SessionPermSetActivationController';

    insert activation;
    return null;
}

public boolean getActivated() {
    Integer alreadyActivated = [SELECT count()
                                FROM SessionPermSetActivation
                                WHERE AuthSessionId = :sessionId
                                And PermissionSetId = :sessionPermSetId LIMIT
                                1];
    return alreadyActivated > 0;
}
}

<apex:page controller="SessionPermSetActivationController">
    <apex:outputPanel rendered="{!!Activated}">
        <h3>Activate Session Permission Set</h3>
        <br />
        <apex:form>
            <apex:commandButton action="{!activate}" value="Activate"
id="activateButton"/>
        </apex:form>
    </apex:outputPanel>
    <apex:outputPanel rendered="{!Activated}">
        <h3>Session Permission Set is already active.</h3>
    </apex:outputPanel>
</apex:page>

```

SetupAuditTrail

Represents changes you or other admins made in your org's Setup area for at least the last 180 days. This object is available in API version 15.0 and later.



Note: SetupAuditTrail is not a supported standard controller. Using SetupAuditTrail as a standard controller in a Visualforce page results in an error.

Supported Calls

`query(), retrieve()`

 **Note:** Aggregate queries aren't supported on this object. For example, `SELECT count() FROM SetupAuditTrail` works but `SELECT count(Id) FROM SetupAuditTrail` fails.

Fields

Field	Details
Action	<p>Type string</p> <p>Properties Filter, Sort</p> <p>Description The category of the change made in Setup. For example, a value of <code>PermSetCreate</code> indicates that an administrator created a permission set. The <code>Display</code> field contains more specific information.</p>
DelegateUser	<p>Type string</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The Login-As user who executed the action in Setup. If a Login-As user didn't perform the action, this field is blank. This field is available in API version 35.0 and later.</p>
Display	<p>Type string</p> <p>Properties Nullable, Sort</p> <p>Description The full description of changes made in Setup. For example, if the <code>Action</code> field has a value of <code>PermSetCreate</code>, the <code>Display</code> field has a value like "Created permission set MAD: with user license Salesforce."</p>
Section	<p>Type string</p> <p>Properties Nullable, Sort</p> <p>Description The section in the Setup menu where the action occurred. For example, Manage Users or Company Profile.</p>



Note: You can use SOQL joins to get the information you need more quickly. For example, running `SELECT CreatedBy.Name FROM SetupAuditTrail LIMIT 10` returns the first and last names of the last 10 people to make changes in Setup.

SetupEntityAccess

Represents the enabled setup entity access settings (such as for Apex classes) for the parent PermissionSet. This object is available in API version 25.0 and later.

To grant users access to an entity, associate the appropriate SetupEntityAccess record with a PermissionSet that's assigned to a user.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the entity's parent PermissionSet.</p>
SetupEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the entity for which access is enabled, such as an Apex class or Visualforce page.</p>
SetupEntityType	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of setup entity for which access is enabled. Valid values are:</p> <ul style="list-style-type: none"> • ApexClass for Apex classes • ApexPage for Visualforce pages • In API version 28.0 and later, ConnectedApplication for OAuth connected apps • In API version 31.0 and later, CustomPermission for custom permissions

Field Name	Details
	<ul style="list-style-type: none"> In API version 28.0 and later, <code>ServiceProvider</code> for service providers In API version 28.0 and later, <code>TabSet</code> for apps

Usage

Because `SetupEntityAccess` is a child of the `PermissionSet` object, the usage is similar to other `PermissionSet` child objects like `FieldPermissions` and `ObjectPermissions`.

For example, the following code returns all permission sets that grant access to any setup entities for which access is enabled:

```
SELECT Id, ParentId, Parent.Name, SetupEntityId
FROM SetupEntityAccess
```

The following code returns permission sets that grant access only to Apex classes:

```
SELECT Id, ParentId, Parent.Name, SetupEntityId
FROM SetupEntityAccess
WHERE SetupEntityType='ApexClass'
```

The following code returns permission sets that grant access to any setup entities, and are not owned by a profile:

```
SELECT Id, ParentId, Parent.Name, SetupEntityId
FROM SetupEntityAccess
WHERE ParentId
IN (SELECT Id
    FROM PermissionSet
    WHERE isOwnedByProfile = false)
```

You may want to return only those permission sets that have access to a specific setup entity. To do this, query the parent object. For example, this code returns all permission sets that grant access to the `helloWorld` Apex class:

```
SELECT Id, Name,
       (SELECT Id, Parent.Name, Parent.Profile.Name
        FROM SetupEntityAccessItems)
  FROM ApexClass
 WHERE Name = 'helloWorld'
```

While it's possible to return permission sets that have access to a `ConnectedApplication`, `ServiceProvider`, or `TabSet` by `SetupEntityId`, it's not possible to return permission sets that have access to these `SetupEntityType` fields by any other `AppMenuItem` attribute, such as `Name` or `Description`. For example, to find out if a user has access to the Recruiting app, you'd run two queries. First, query to get the `AppMenuItem` ID:

```
SELECT Id, Name, Label
FROM AppMenuItem
WHERE Name = 'Recruiting'
```

Let's say the previous query returned the `AppMenuItem` ID `02uD0000000GiMIAW`. Using this ID, you can now run a query to find out if a user has access to the Recruiting app:

```
SELECT Id, SetupEntityId, SetupEntityType
FROM SetupEntityAccess
WHERE ParentId
IN
```

```
(SELECT PermissionSetId
  FROM PermissionSetAssignment
 WHERE AssigneeId = '005D0000001QOzF')
AND (SetupEntityId = '02uD000000GIiMIAW')
```

SEE ALSO:

[PermissionSet](#)
[FieldPermissions](#)
[ObjectPermissions](#)
[ApexClass](#)
[ApexPage](#)

SignupRequest

Represents a request for a new Trialforce sign-up. This object is available in API version 27.0 and later.

 **Note:** You are limited to 20 sign-ups per day. If you need to make additional sign-ups, log a case in the Partner Community.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete()`

Fields

Field Name	Details
AuthCode	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A one-time authorization code that can be exchanged for an OAuth access token and refresh token using standard Salesforce APIs. It's used in conjunction with <code>ConnectedAppCallbackUrl</code> and <code>ConnectedAppConsumerKey</code>, when the specified connected app hasn't been configured with an X.509 certificate. This is a read-only field provided by the system once the sign-up request has been processed. This field is available in API version 29.0 and later.</p>
Company	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description The name of the company requesting the trial sign-up.</p>
ConnectedAppCallbackUrl	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description When used in conjunction with <code>ConnectedAppConsumerKey</code>, specifies a connected app that should be approved automatically during the sign-up creation. This field is available in API version 28.0 and later.</p>
ConnectedAppConsumerKey	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description When used in conjunction with <code>ConnectedAppCallbackUrl</code>, specifies a connected app that should be approved automatically during the sign-up creation. This field is available in API version 28.0 and later.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The two-character, upper-case ISO-3166 country code. You can find a full list of these codes at a number of sites, such as: www.iso.ch/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html. The language of the trial organization is auto-determined based on the value of this field.</p>
CreatedOrgId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 15-character organization ID of the trial organization created. This is a read-only field provided by the system once the sign-up request has been processed.</p>
CreatedOrgInstance	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The server instance of the new trial organization, for example, "na8." This field is available in API version 29.0 and later.</p>
Edition	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The Salesforce template that is used to create the trial organization. Possible values are Partner Group, Professional, Partner Professional, Sales Professional, Professional TSO, Enterprise, Partner Enterprise, Service Enterprise, Enterprise TSO, Developer, and Partner Developer. This field is available in API version 35.0 and later.</p>
ErrorCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The error code if the sign-up request isn't successful. This is a read-only field provided by the system to be used for support purposes.</p>
FirstName	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort</p> <p>Description The first name of the admin user for the trial sign-up.</p>
LastName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The last name of the admin user for the trial sign-up.</p>
PreferredLanguage	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort</p>
	<p>Description The language of the trial organization being created. Specify the language using a language code listed under Fully Supported Languages in "Which Languages Does Salesforce Support?" in the Salesforce Help. For example, use <code>zh_CN</code> for simplified Chinese. The value you select overrides the language set by locale. If you specify an invalid language, the organization defaults to English. Likewise, if you specify a language that isn't supported by the Salesforce edition associated with your trial template, the trial organization defaults to English. This field is available in API version 35.0 and later.</p>
ResolvedTemplateId	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description Populated during the sign-up request and for internal use by Salesforce. This field is available in API version 35.0 and later.</p>
ShouldConnectToEnvHub	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description When set to <code>true</code>, the trial organization is connected to the Environment Hub. The sign-up must take place in the hub master organization or a spoke organization. This field is available in API version 35.0 and later.</p>
SignupEmail	<p>Type email</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The email address of the admin user for the trial sign-up.</p>
SignupSource	<p>Type string</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description A user-specified description of the trial sign-up, up to 60 characters in length. This field is available in API version 36.0 and later.</p>

Field Name	Details
Status	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The status of the request. Possible values are New, In Progress, Error, or Success. The default value is New.</p>
Subdomain	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The subdomain for the new trial organization when it uses a custom My Domain. The maximum length is 33 characters for Developer Edition (DE) and 40 characters for all other editions (because a suffix is appended to all DE organizations).</p>
SuppressSignupEmails	<p>Type boolean</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When set to true, no sign-up emails are sent when the trial organization is created. This field is used for the Proxy Signup feature, and is available in API version 29.0 and later.</p>
TemplateId	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The 15-character ID of the Trialforce template that is the basis for the trial sign-up. The template must be approved by Salesforce. If you don't specify an edition, a template ID is required.</p>
TrialDays	<p>Type anyType</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description The duration of the trial sign-up in days. Must be equal to or less than the trial days for the approved Trialforce template. If not provided, it defaults to the trial duration specified for the Trialforce template.</p>

Field Name	Details
TrialSourceOrgId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 15-character organization ID of the Trialforce Source Organization from which the Trialforce template was created.</p>
Username	<p>Type string</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The username of the admin user for the trial sign-up. It must follow the address convention specified in RFC822: www.w3.org/Protocols/rfc822/#z10</p>

Usage

The Java class below uses the REST API to create a SignupRequest object. It authenticates to the Trialforce Management Organization and then posts a request to the SignupRequest object.

Here are the variables you need to specify in this example.

- SERVER — The name of the host server for the Trialforce Management Organization (TMO), for example, "*yourInstance.salesforce.com*".
- USERNAME — The admin username for the TMO.
- PASSWORD — The concatenation of the admin password and the security token for the TMO. To get an email with the security token, from your personal settings in Salesforce select **Reset My Security Token** and click **Reset Security Token**.
- CLIENT_ID — From Setup in Salesforce, enter *Apps* in the Quick Find box, select **Apps**, and click **New** under Connected Apps. Enter values for the required fields (the Callback URL is required but can initially be set to any valid URL as it's not used), grant full access for the OAuth scopes in the "Selected OAuth Scopes" selector, and click **Save**. Then copy the value of "Consumer Key" and use it for this variable.
- CLIENT_SECRET — On the same page, click **Click to reveal**. Then copy the value of "Consumer Secret" and use it for this variable.

```
public class IsvSignupDriver {
    private static final String SERVER = server_name:port;
    private static final String USERNAME = tmo_username;
    private static final String PASSWORD = tmo_passwordsecurity_token;
    private static final String CLIENT_ID = consumer_key;
    private static final String CLIENT_SECRET = consumer_secret;

    private static SignupRequestInfo signupRequest = null;

    public static String createSignupRequest (SignupRequestInfo sr)
        throws JSONException, IOException {
        JSONObject createResponse = null;
```

```

signupRequest = sr;
JSONObject loginResponse = login(SERVER, USERNAME, PASSWORD);
String instanceUrl = loginResponse.getString("instance_url");
String accessToken = loginResponse.getString("access_token");
createResponse = create(instanceUrl, accessToken);
System.out.println("Created SignupRequest object: " + createResponse + "\n");
return createResponse.toString();
}

/* Authenticates to the TMO using the required credentials */

private static JSONObject login(String server, String username, String password)
throws ClientProtocolException, IOException, JSONException {
String authEndPoint = server + "/services/oauth2/token";
HttpClient httpclient = new DefaultHttpClient();
try {
HttpPost post = new HttpPost(authEndPoint);

List<NameValuePair> params = new ArrayList<NameValuePair>();
params.add(new BasicNameValuePair("grant_type", "password"));
params.add(new BasicNameValuePair("client_id", CLIENT_ID));
params.add(new BasicNameValuePair("client_secret", CLIENT_SECRET));
params.add(new BasicNameValuePair("username", username));
params.add(new BasicNameValuePair("password", password));
post.setEntity(new UrlEncodedFormEntity(params, Consts.UTF_8));

BasicResponseHandler handler = new BasicResponseHandler();
String response = httpclient.execute(post, handler);
return new JSONObject(response);
} finally {
httpclient.getConnectionManager().shutdown();
}
}
/* Posts a request to the SignupRequest object */

private static JSONObject create(String instanceUrl, String accessToken)
throws ClientProtocolException, IOException, JSONException {
HttpClient httpClient = new DefaultHttpClient();
try {
HttpPost post = new HttpPost(instanceUrl +
"/services/data/v27.0/sobjects/SignupRequest/");
post.setHeader("Authorization", "Bearer " + accessToken);
post.setHeader("Content-Type", "application/json");

JSONObject requestBody = new JSONObject();
requestBody.put("TemplateId", signupRequest.getTemplateID());
requestBody.put("SignupEmail", signupRequest.getEmail());
requestBody.put("username", signupRequest.getUsername());
requestBody.put("Country", "US");
requestBody.put("Company", signupRequest.getCompanyName());
requestBody.put("lastName", signupRequest.getLastName());

StringEntity entity = new StringEntity(requestBody.toString());
post.setEntity(entity);
}
}

```

```
        BasicResponseHandler handler = new BasicResponseHandler();
        String response = httpClient.execute(post, handler);
        return new JSONObject(response);
    } finally {
        httpClient.getConnectionManager().shutdown();
    }
}
```

Error Codes

If the sign-up fails, the system generates an error code that can help you identify the cause. This table shows the most important error codes.

Error Code	Description
C-1007	Duplicate username.
C-1015	Error while establishing the new organization's My Domain settings. Contact Salesforce support for assistance.
C-1016	Error while configuring the OAuth connected app for Proxy Signup. Verify that your connected app has a valid consumer key, callback URL, and unexpired certificate (if applicable).
C-1018	Invalid subdomain value provided during sign-up.
C-1019	Subdomain in use. Please choose a new subdomain value.
C-9999	Generic "fatal error." Contact Salesforce support for assistance.
S-1006	Invalid email address (not in a proper email address format).
S-2006	Invalid country code.
T-0001	Template ID not valid (not in the format OTxxxxxxxxxx).
T-0002	Template not found. Either the template doesn't exist (it may have been deleted), or it doesn't exist at the appropriate version.
T-0003	Template not approved for use by Salesforce.

Site

Represents a public website that is integrated with an Organization. This object is generally available in API version 16.0 and later.
To access this object, Force.com Sites or Site.com must be enabled for your organization.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

- Customer Portal users can't access this object.
- To view this object, you must have the "View Setup and Configuration" permission.

Fields

Field	Description
AdminId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The site administrator designated as the contact for the site. This user receives site-related communications from site visitors and from Salesforce.</p>
AnalyticsTrackingCode	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The tracking code associated with your site. This code can be used by services like Google Analytics to track page request data for your site.</p>
ClickjackProtectionLevel	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description Sets the clickjack protection level. The options are:</p> <ul style="list-style-type: none"> AllowAllFraming—Allow framing by any page (no protection) SameOriginOnly—Allow framing by the same origin only (recommended) NoFraming—Don't allow framing by any page (most protection) <p>This field is available in API version 30.0 and later.</p>
DailyBandwidthLimit	<p>Type int</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The rolling 24-hour daily bandwidth limit for the sites in your organization.</p>

Field	Description
DailyBandwidthUsed	<p>Type double</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The current rolling 24-hour daily bandwidth usage for the sites in your organization.</p>
DailyRequestTimeLimit	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The rolling 24-hour daily service request time limit for the sites in your organization. Service request time is calculated as the total server time in minutes required to generate pages for the site.</p>
DailyRequestTimeUsed	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The current rolling 24-hour daily service request time for the sites in your organization.</p>
Description	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort</p> <p>Description An optional description of the site.</p>
GuestUserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The site or Salesforce Communities specific user that anonymous, unauthenticated users run as when interacting with the site.</p>
MasterLabel	<p>Type string</p>

Field	Description
Properties	Filter, Group, Sort
Description	The name of the site as it appears in the user interface.
MonthlyPageViewsEntitlement	Type
int	Properties
Filter, Group, Nillable, Sort	Description
The number of page views allowed for the current calendar month for the sites in your organization.	
Name	Type
string	Properties
Filter, Group, Sort	Description
The name used when referencing the site in the API.	
OptionsAllowHomePage	Type
boolean	Properties
Filter	Description
The option to enable the standard page associated with the Home tab (/home/home.jsp).	
OptionsAllowStandardAnswersPages	Type
boolean	Properties
Filter	Description
The option to enable standard pages associated with an answers community. If you want to use default Answers pages (such as AnswersHome), enable these pages.	
OptionsAllowStandardIdeasPages	Type
boolean	Properties
Filter	

Field	Description
Description	The option to enable standard pages associated with an Ideas community. If you want to use default Ideas pages (such as IdeasHome), enable these pages.
OptionsAllowStandardLookups	<p>Type boolean</p> <p>Properties Filter</p> <p>Description The option to enable the standard lookup pages. These are the popup windows associated with lookup fields on Visualforce pages.</p>
OptionsAllowStandardSearch	<p>Type boolean</p> <p>Properties Filter</p> <p>Description The option to enable the standard search pages. To allow public users to perform standard searches, enable these pages.</p>
OptionsEnableFeeds	<p>Type boolean</p> <p>Properties Filter</p> <p>Description The option that displays the Syndication Feeds related list, where you can create and manage syndication feeds for users on your public sites. This field is visible only if you have the feature enabled for your organization.</p>
OptionsRequireHttps	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When true, the site requires secure connections. When false, the site operates normally via insecure connections instead of redirecting to a secure connection.</p>
SiteType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field	Description
	Description
	<p>Identifies whether the site is a Visualforce (Force.com Sites) or a Site.com site. <code>SiteType</code> is available in API version 21.0 and later. In API version 26.0 and later, if Salesforce Communities is enabled for your organization, the site could also be a Network Visualforce or Network Site.com site.</p>
Status	Type picklist
	Properties
	Filter, Group, Restricted picklist, Sort
	Description
	The status for the site. For example, <code>Active</code> or <code>In Maintenance</code> .
Subdomain	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The Force.com domain that you registered for your site. For example, if your Force.com domain is <code>mycompany.force.com</code> , then <code>mycompany</code> is the subdomain.
TopLevelDomain	Type url
	Properties
	Filter, Nillable
	Description
	The optional branded custom Web address that you registered with a third-party domain name registrar. The custom Web address acts as an alias to your Force.com address.
	Beginning with API version 21.0, <code>TopLevelDomain</code> is no longer available. Instead, use the Domain and DomainSite objects.
UrlPathPrefix	Type string
	Properties
	Filter, Group, Nillable, Sort
	Description
	The unique Force.com URL that the public uses to access this site.

Usage

Use this read-only object to query or retrieve information on your Force.com site.

SiteDomain

SiteDomain is a read-only object, and a one-to-many replacement for the Site.TopLevelDomain field. This object is available in API version 21.0, and has been deprecated as of API version 26.0. In API version 26.0 and later, use the [Domain](#) and [DomainSite](#) objects instead.

To access this object, Force.com Sites or Site.com must be enabled for your organization.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

- Customer Portal users can't access this object.
- To view this object, you must have the "View Setup and Configuration" permission.

Fields

Field	Description
Domain	<p>Type url</p> <p>Properties Filter, Sort</p> <p>Description The branded custom Web address within the global namespace identified by this domain's type. In the Domain Name System (DNS) global namespace, this field is the custom Web address that you registered with a third-party domain name registrar. The custom Web address can be used to access the site of this domain.</p>
SiteId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the associated Site.</p>
DomainType	<p>Type picklist</p>

Field	Description
	<p>Properties Filter, Group, Sort, Nillable</p> <p>Description The global namespace that this custom Web address belongs to. This value is set to DNS for custom Web addresses in the global DNS. This field is available in version 24.0 of the API.</p>

Usage

Use this read-only object to query the custom Web addresses that are associated with each website in your organization.

SiteHistory

Represents the history of changes to the values in the fields of a site. This object is generally available in API version 18.0 and later.

To access this object, Force.com Sites must be enabled for your organization.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

- Customer Portal users can't access this object.
- To view this object, you must have the "View Setup and Configuration" permission.

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field	Details
	<p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The last value of the field before it was changed.</p>
SiteId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the associated Site.</p>

Skill

Represents a category or group that Live Agent users or field service resources can be assigned to. This object is available in API version 24.0 and later.

 **Note:** For information about Work.com skills on a user's profile, see the [ProfileSkill](#) topic.

Supported Calls

`create(), describeSObjects(), query(), retrieve(), search(), update(), upsert()`

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the skill.</p>
DeveloperName	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the skill.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed the skill.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update, idLookup</p> <p>Description The name of the skill.</p>

Usage

Live Agent

Use this object to assign Live Agent users to groups based on their abilities. The skills associated with a LiveChatButton determine which agents receive chat requests that come in through that button.

Field Service Lightning

Use this object to track certifications and areas of expertise in your workforce. After you create a skill, you can:

- Assign it to a service resource via the Skills related list on the resource's detail page. When you assign a skill to a service resource, you can specify their skill level and the duration of the skill.
- Add it as a required skill via the Skill Requirements related list on any work type, work order, or work order line item. When you add a required skill to a work record, you can specify the skill level.

SkillProfile

Represents a join between Skill and Profile. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
ProfileId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the profile.</p>
SkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the skill.</p>

Usage

Use this object to assign specific skills to specific profiles.

SkillRequirement

Represents a skill that is required to complete a particular task. Skill requirements can be added to work types, work orders, and work order line items. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeLayout(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Fields

Field Name	Details
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The record that the skill is required for. The related record can be a work order, work order line item, or work type.</p>
SkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The skill that is required.</p>
SkillLevel	<p>Type double</p>

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Nillable, Sort, Update</p> <p>Description The level of the skill required. Skill levels can range from zero to 99.99. Depending on your business needs, you may want the skill level to reflect years of experience, certification levels, or license classes.</p>
SkillNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description An auto-generated number identifying the skill requirement.</p>

Usage

If a work order or work order line item can only be completed by a service resource with a particular set of skills, add *required skills* to the record via the Skill Requirements related list. Required skills help dispatchers assign the record to a service resource with the proper expertise. You can still assign a work order, work order line item, or related service appointment to a service resource that does *not* possess the required skills, so required skills serve more as a suggestion than a rule.



Note: If you're using the Field Service Lightning managed packages, you can use matching rules to ensure that appointments are only assigned to service resources who possess the required skills listed on the parent work order.

If many of your work orders require the same skills, we recommend adding required skills to *work types* to save yourself some time and keep your processes consistent. When you add a required skill to a work type, work orders and work order line items that use that type automatically inherit the required skill. For example, if you know that all annual maintenance visits for your Classic Refrigerator product require a Refrigerator Maintenance skill level of at least 50, you can add that required skill to the Annual Maintenance Visit work type. When it's time to create a work order for a customer's annual fridge maintenance, adding that work type to the work order adds the required skill as well.

SkillRequirementFeed

Represents a single feed item on a skill requirement record detail page. This object is available in API version 38.0 and later.

A skill requirement feed shows changes to tracked fields on a skill requirement record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to skill requirements in Salesforce.

Supported Calls

`delete(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Operating Hours object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>

Field Name	Details
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the skill requirement record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	Description
	<p>The type of feed item.</p>

SkillRequirementHistory

Represents the history of changes made to tracked fields on a skill requirement. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>

Field Name	Details
SkillRequirementId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the skill requirement being tracked. The history is displayed on the detail page for this record.</p>

SkillUser

Represents a join between Skill and User. This object is available in API version 24.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
SkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the skill.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the user.</p>

Usage

Use this object to assign specific skills to specific users.

SlaProcess

Represents an entitlement process associated with an Entitlement. This object is available in API version 19.0 and later.

An entitlement process is a timeline that includes all the steps (MilestoneType records) that your support team must complete to resolve cases. Each process includes the logic necessary to determine how to enforce the correct service level for your customers.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

All users, even those without the "View Setup and Configuration" user permission, can view entitlement processes via the API.

Fields

Field	Details
BusinessHoursId	Type reference Properties Filter, Group, Nillable, Sort Description Required. ID of the BusinessHours associated with the entitlement. Must be a valid business hours ID.
Description	Type textarea Properties Filter, Nillable Description A description of the entitlement process.
IsActive	Type boolean Properties Defaulted on create, Filter Description Indicates whether the entitlement process is active (<code>true</code>) or not (<code>false</code>).
IsVersionDefault	Type boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the entitlement process is the default version (<code>true</code>) or not (<code>false</code>). This field is available in API version 28.0 and later in organizations that have entitlement versioning enabled.</p>
Name	<p>Type string</p> <p>Properties Filter, idLookup</p> <p>Description The name of the entitlement process.</p>
NameNorm	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The read-only value for the unique name of the entitlement process or the entitlement process version. If entitlement versioning is enabled, this value is automatically generated for each version of an entitlement process in this form: <code>process name+v+x</code>, where <code>x</code> is the version number (for example, "gold_support_v2"). If entitlement versioning isn't enabled, this value is the same as Name. This field is available in API version 28.0 and later.</p>
SObjectType	<p>Type picklist</p> <p>Properties Restricted picklist, Filter, Group, Sort</p> <p>Description The type of records that the entitlement process can run on. Its values are:</p> <ul style="list-style-type: none"> • Case • Work Order <p>An entitlement process runs only on records that match its type. For example, a Case entitlement process that's applied to an entitlement runs only on cases associated with the entitlement, not on work orders. As a best practice, therefore, manage customers' work orders and cases on separate entitlements.</p> <p>The field label in the user interface is Entitlement Process Type.</p>

Field	Details
StartDateField	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The criteria for cases to enter the entitlement process. Cases can enter the process based on:<ul style="list-style-type: none">• The creation date on a case• A custom date/time field on a case</p>
VersionMaster	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Identifies the sequence of versions to which this entitlement process belongs. This field's contents can be any value as long as it is identical among all versions of the entitlement process. This field is available in API version 28.0 and later in organizations that have entitlement versioning enabled.</p>
VersionNotes	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The description of the entitlement process version. This field is available in API version 28.0 and later in organizations that have entitlement versioning enabled.</p>
VersionNumber	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The version number of the entitlement process. Must be 1 or greater. This field is available in API version 28.0 and later in organizations that have entitlement versioning enabled.</p>

Usage

Use this object to query entitlement processes on entitlements.

SEE ALSO:

[Entitlement](#)
[MilestoneType](#)
[CaseMilestone](#)

Solution

Represents a detailed description of a customer issue and the resolution of that issue.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted .
IsHtml	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the Solution is an HTML solution (<code>true</code>) or not (<code>false</code>).
IsOutOfDate	Type boolean Properties Defaulted on create, Filter

Field	Details
	<p>Description Read-only field that indicates whether a solution master has been updated since the translated version was created (<code>true</code>) or not (<code>false</code>). Note that this field does not appear in the page layout of master solutions.</p>
IsPublished	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the Solution has been published (<code>true</code>) or not (<code>false</code>). A solution's published state does not affect how it can be used, or whether you can query, update, or delete it. Label is Public.</p> <p> Note: Prior to Spring '14, the label was Visible in Self-Service Portal.</p>
IsPublishedInPublicKb	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the Solution has been published as a Public Solution (<code>true</code>) or not (<code>false</code>). Label is Visible in Public Knowledge Base. This field only applies to solutions, not articles in the public knowledge base.</p>
IsReviewed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the Solution has been reviewed (<code>true</code>) or not (<code>false</code>). This flag can only be set indirectly via the <code>status</code> picklist. Each predefined <code>status</code> value implies an <code>IsReviewed</code> value. Label is Reviewed.</p>
LastReferencedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>

Field	Details
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the User who owns the Solution.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description ID of the master solution, if this is the translation of a master solution.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description ID of the RecordType to which the Solution is associated.</p>
SolutionLanguage	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description The language that the solution is written in, such as French or Chinese (Traditional).</p>
SolutionName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Required. If a client application creates a new Solution and a value for this field is unspecified, a hyphen (-), the default value for this field, is used. Limit: 255 characters. Label is Title.</p>
SolutionNote	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description The details of the Solution record. Limit: 32,000 characters. Label is Solution Details.</p> <p> Note: If you have HTML Solutions enabled, any HTML tags used in this field are verified before the object is created or updated. If invalid HTML is entered, an error is thrown. Any JavaScript used in this field is removed before the object is created or updated.</p>
SolutionNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description An identifying number that is assigned automatically when a solution is created. It can't be set directly, and it can't be modified.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Required. The status of the solution. Directly controls the <code>IsReviewed</code> value. To obtain the status values in the picklist, a client application can query the <code>SolutionStatus</code>.</p>
TimesUsed	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of times this solution has been used. Label is Num Related Case.</p>

Usage

Use this object to manage your organization's solutions. Client applications can create, update, delete, and query Attachment records associated with a solution.

SEE ALSO:

[CategoryData](#)

[CategoryNode](#)

SolutionFeed

Represents a single feed item in the feed displayed on the detail page for a solution record. This object is available in API version 18.0 and later.

A solution feed shows recent changes to a solution record for any fields that are tracked in feeds, and comments and posts about the record. It is a useful way to stay up-to-date with changes to solutions.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Solution object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the SolutionFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>

Field Name	Details
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission. This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only.</p>

Field Name	Details
ContentFileName	<p>The description of the file specified in <code>ContentData</code>.</p> <p>Type string</p> <p>Properties <code>Group</code>, <code>Nillable</code>, <code>Sort</code></p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if <code>Type</code> is <code>ContentPost</code>. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type <code>int</code></p> <p>Properties <code>Group</code>, <code>Nillable</code>, <code>Sort</code></p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties <code>Group</code>, <code>Nillable</code>, <code>Sort</code></p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
FeedPostId	<p>Type reference</p> <p>Properties <code>Filter</code>, <code>Group</code>, <code>Nillable</code>, <code>Sort</code></p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. Use <code>FeedItem</code> instead. The ID of the associated FeedPost. A FeedPost represents the following types of changes in a SolutionFeed: status updates, changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties <code>Group</code>, <code>Nillable</code>, <code>Sort</code></p>

Field Name	Details
	<p>Description</p> <p>The ID of the user who added this object to the feed. For example, if a client application migrates multiple posts and comments from another application into the feed, <code>InsertedById</code> is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description</p> <p>This field is a standard system field.</p>

Field Name	Details
	<p>When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a <code>FeedComment</code> is inserted on the feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for the <code>FeedComment</code>. Deleting the <code>FeedComment</code> does not change the <code>LastModifiedDate</code>.</p> <p>Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of the <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none">• <code>NetworkId</code>—The ID of the community in which the <code>FeedItem</code> is available. If left empty, the feed item is only available in the default community.• <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>.• For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>.• You can't filter a <code>FeedItem</code> on the <code>NetworkScope</code> field.

Field Name	Details
ParentId	Type reference
	Properties Filter, Group, Sort
	Description
	ID of the solution record that is tracked in the feed. The feed is displayed on the detail page for this record.
RelatedRecordId	Type reference
	Properties Group, Nillable, Sort
	Description
	ID of the ContentVersion article associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a SolutionFeed object of Type ContentPost.
Title	Type string
	Properties Group, Nillable, Sort
	Description
	The title of the SolutionFeed. When the Type is LinkPost, the LinkUrl is the URL, and this field is the link name.
Type	Type picklist
	Properties Filter, Group, Nillable, Restricted picklist, Sort
	Description
	The type of SolutionFeed item:
	<ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).
	For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.
	<ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement.

Field Name	Details
	<ul style="list-style-type: none">● <code>AnnouncementPost</code>—Not used.● <code>ApprovalPost</code>—generated when a user submits an approval.● <code>BasicTemplateFeedItem</code>—Not used.● <code>CanvasPost</code>—a post made by a canvas app posted on a feed.● <code>CollaborationGroupCreated</code>—generated when a user creates a public group.● <code>CollaborationGroupUnarchived</code>—Not used.● <code>ContentPost</code>—a post with an attached file.● <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher.● <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold.● <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed.● <code>LinkPost</code>—a post with an attached URL.● <code>PollPost</code>—a poll posted on a feed.● <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile.● <code>QuestionPost</code>—generated when a user posts a question.● <code>ReplyPost</code>—generated when Chatter Answers posts a reply.● <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com.● <code>TextPost</code>—a direct text entry on a feed.● <code>TrackedChange</code>—a change or group of changes to a tracked field.● <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated.

The following values appear in the `Type` picklist for all feed objects but apply only to [CaseFeed](#):

- `AttachArticleEvent`—generated event when a user attaches an article to a case.
- `CallLogPost`—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.
- `CaseCommentPost`—generated event when a user adds a case comment for a case object.
- `ChangeStatusPost`—generated event when a user changes the status of a case.
- `ChatTranscriptPost`—generated event when Live Agent transcript is saved to a case.
- `EmailMessageEvent`—generated event when an email related to a case object is sent or received.

Field Name	Details
	<ul style="list-style-type: none">• <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated.• <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status.• <code>SocialPost</code>—generated when a social post is created from a case.
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• <code>Visibility</code> can be updated on record posts.• The <code>update</code> property is supported only for feed items posted on records.

Usage

Use this object to track changes for a solution record.

SEE ALSO:

[Solution](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

SolutionHistory

Represents the history of changes to the values in the fields of a solution.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed. Label is Custom Field Definition ID.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed. Limited to 255 characters.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed. Limited to 255 characters.</p>
SolutionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Solution. Label is Solution ID.</p>

Usage

Use this read-only object to identify changes to a solution.

This object respects field-level security on the parent object.

SEE ALSO:

[SolutionStatus](#)

SolutionStatus

Represents the status of a Solution, such as Draft, Reviewed, and so on.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, NillableSort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
IsDefault	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the default solution status value (<code>true</code>) or not (<code>false</code>) in the picklist. Only one value can be the default value.</p>
IsReviewed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this solution status value represents a reviewed Solution (<code>true</code>) or not (<code>false</code>). Multiple solution status values can represent a reviewed Solution.</p>

Field	Details
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this solution status value. This display value is the internal label that does not get translated.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the solution status picklist. These numbers are not guaranteed to be sequential, as some previous solution status values might have been deleted.</p>

Usage

This object represents a value in the solution status picklist. The solution status picklist provides additional information about the status of a Solution, such as whether a given status value represents a reviewed or unreviewed solution. Your client application can query this object to retrieve the set of values in the solution status picklist, and then use that information while processing Solution objects to determine more information about a given solution. For example, the application could test whether a given case has been reviewed or not based on its `Status` value and the value of the `IsReviewed` property in the associated SolutionStatus record.

SEE ALSO:

[Solution](#)

SolutionTag

Associates a word or short phrase with a Solution.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the tagged item.</p>
Name	<p>Type string</p> <p>Properties Create, Filter</p> <p>Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.</p>
TagDefinitionId	<p>Type reference</p> <p>Properties Filter</p> <p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag. Valid values:<ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId. </p>

Usage

SolutionTag stores the relationship between its parent TagDefinition and the Solution being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

SOSDeployment

Represents the general settings for deploying SOS video call capability in a native mobile application. This object is available in API version 34.0 and later.

Supported Calls

`create(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the deployment.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description The name of the deployment.</p>
OptionsIsBackwardFacingCameraEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether customers can use the backwards-facing camera on their mobile devices to talk to SOS agents.</p>
OptionsIsEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether the deployment is enabled for customers to request new SOS video calls.</p>
OptionsIsVoiceOnlyMode	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether video functionality is disabled for customers, making it so customers can only talk to SOS agents using only audio.</p>
QueueId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the queue that's associated with the SOS deployment.</p>

Usage

Use this object to query and manage SOS deployments.

SOSSession

This object is automatically created for each SOS session and stores information about the session. This object is available in API versions 34.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, ~~search()~~, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
AppVersion	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The version of the customer's mobile application in which SOS is implemented.</p>
CaseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the case that's associated with the SOS session.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the contact that's associated with the SOS session.</p>
DeploymentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the SOS deployment that the SOS session originated from.</p>
EndTime	<p>Type dateTime</p>

Field Name	Details
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The time that the SOS session ended.
IpAddress	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The IP address from which the customer's SOS call originated.
LastReferencedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date and time that the session record was last referenced by a user.
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The date and time that the session record was last viewed.
Name	Type
	string
	Properties
	Autonumber, Defaulted on create, Filter, Sort
	Description
	The name of the session.
OpentokSession	Type
	encryptedstring
	Properties
	Create, Nillable, Update
	Description
	The ID of the OpenTok session that's associated with the SOS video call.
OwnerId	Type
	reference

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>
	<p>Description The ID of the session record's owner.</p>
SessionDuration	<p>Type int</p>
	<p>Properties Filter, Group, Nillable, Sort</p>
	<p>Description The amount of time that the SOS session lasted.</p>
SessionRecordingUrl	<p>Type url</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The URL where the SOS session recording is stored.</p>
SosVersion	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description The version of SOS that was used in your organization's mobile application when this session occurred.</p>
StartTime	<p>Type dateTime</p>
	<p>Properties Create, Filter, Nillable, Sort, Update</p>
	<p>Description The time that the SOS session began.</p>
SystemInfo	<p>Type string</p>
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p>
	<p>Description Information about the customer's mobile device from which the SOS call originated, such as the device's operating system.</p>

Field Name	Details
WaitDuration	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time the customer waited before an agent accepted the SOS session and the call began.</p>

Usage

Use this object to query and manage SOS session records.

SOSSessionActivity

Captures information about specific events that occur during an SOS video call, such as when an SOS call begins or ends. This object is available in API version 34.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
ActivityTime	<p>Type dateTime</p> <p>Properties Create, Filter, Sort</p> <p>Description The time at which the activity occurred.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, idLookup, Filter, Sort</p> <p>Description The name of the activity.</p>
SessionId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the SOS session that's associated with the event.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The kind of activity that occurred.</p>

Usage

Use this object to query and manage SOS session activities.

SOSSessionHistory

This object is automatically created for each SOS session and stores information about changes made to the session. This object is available in API versions 34.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed in a session record.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	<p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The original value of the field that was changed.</p>
SOSSessionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the session record that was changed.</p>

Usage

Use this object to identify changes to SOS session records.

SOSSessionOwnerSharingRule

Represents the rules for sharing an SOS session record with users other than the record owner. This object is available in API version 34.0 and later.



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description Determines the level of access users have to session records. Specifies whether or not users can read, edit, or transfer session records. Corresponds to the <code>Default Access</code> column in the UI.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum length is 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to <code>Rule Name</code> in the UI.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID that represents the source group. Session records that are owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p>

Field Name	Details
	<p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the UI.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID that represents the User or Group that you are granting access to.</p>

Usage

Use this object to manage the sharing rules for SOS session records.

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

SOSSessionShare

Represents a sharing entry on an SOS session. This object is available in API version 34.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the SOSSession. The possible values are:</p> <ul style="list-style-type: none"> • Read

Field Name	Details
	<ul style="list-style-type: none"> • <code>Edit</code> • <code>All</code> (This value is not valid for <code>create()</code> or <code>update()</code> calls.)
	<p>This field must be set to an access level that is higher than the organization's default access level for live chat transcripts.</p>
<code>ParentId</code>	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description ID of the parent object, if any.</p>
<code>RowCause</code>	<p>Type picklist</p>
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p>
	<p>Description Reason that this sharing entry exists.</p>
	<p>You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>
	<p>Values can include:</p>
	<ul style="list-style-type: none"> • <code>Manual</code>—The user or group has access because a user with "All" access manually shared the SOSSession with them. • <code>Owner</code>—The user is the owner of the SOSSession or is in a role above the SOSSession owner in the role hierarchy.
<code>UserOrGroupId</code>	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description ID of the user or group that has been given access to the SOSSession.</p>

Usage

This object lets you determine which users and groups can view and edit SOSSession records that are owned by other users.

If you attempt to create a new record that matches an existing record, the `create()` call updates any modified fields and returns the existing record.

StaticResource

Represents a static resource that can be used in Visualforce markup.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Fields

Field	Details
Body	<p>Type base64</p> <p>Properties Create, Nillable, Update</p> <p>Description Required. Encoded file data.</p>
BodyLength	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Size of the file (in bytes).</p>
CacheControl	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The sharing policy for the static resource when cached. The cache control can have one of the following values:</p> <ul style="list-style-type: none"> • <i>Private</i> specifies that the static resource data cached on the Salesforce server shouldn't be shared with other users. The static resource is only stored in cache for the current user's session. • <i>Public</i> specifies that the static resource data cached on the Salesforce server be shared with other users in your organization for faster load times.
ContentType	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Type of content. Label is Mime Type. Limit: 120 characters.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Text description of the static resource. Limit: 255 characters.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of the static resource.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>

Usage

Use static resources to upload content that you can reference in Visualforce markup, including archives (such as .zip and .jar files), images, stylesheets, JavaScript, and other files. Using a static resource is preferable to uploading a file to the Documents tab because:

- You can package a collection of related files into a directory hierarchy and upload that hierarchy as a .zip or .jar archive.
- You can reference a static resource in page markup by name using the `$Resource` global variable instead of hard-coding document IDs.

Encoded Data

The API sends and receives the binary file data encoded as a base64 data type. Prior to creating a record, clients must encode the binary file data as base64. Upon receiving an API response, clients must decode the base64 data to binary (this conversion is usually handled for you by the SOAP client).

Maximum Static Resource Size

You can create or update static resources to a maximum size of 5 MB. An organization can have up to 250 MB of static resources, total.

SEE ALSO:

[ApexComponent](#)

[ApexPage](#)

[Developer Guide: Visualforce Developer Guide](#)

StreamingChannel

Represents a channel that is the basis for notifying listeners of generic Streaming API events. Available from API version 29.0 or later.

Supported Calls

REST: DELETE, GET, PATCH, POST (query requests are specified in the URI)

SOAP: `create()`, `delete()`, `describe()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`

Special Access Rules

- This object is available only if Streaming API is enabled for your organization.
- Only users with “Create” permission can create this record.
- You can create a permission set and grant users read and create access to all streaming channels in the org. This access isn’t for a specific channel, like with user sharing.
- You can apply user sharing to `StreamingChannel`. You can restrict access to receiving or sending events on a channel by sharing channels with specific users or groups. Channels shared with public read-only or read-write access send events only to clients subscribed to the channel that also are using a user session associated with the set of shared users or groups. Only users with read-write access to a shared channel can generate events on the channel, or modify the actual `StreamingChannel` record.

Dynamic Streaming Channel

Generic Streaming also supports dynamic streaming channel creation, which creates a StreamingChannel when a client first subscribes to the channel. To enable dynamic streaming channels in your org, from Setup, enter *User Interface* in the Quick Find box, then select **User Interface** and enable **Enable Dynamic Streaming Channel Creation**.

Fields

Field	Field Type	Description
Description	string	<p>Description of the StreamingChannel. Limit: 255 characters.</p> <p>Field Properties: Create, Filter, Group, Nillable, Sort, Update</p> <p>Label: Description</p>
ID	ID	<p>System field: Globally unique string that identifies a StreamingChannel record.</p> <p>Field Properties: Default on create, Filter, Group, idLookup, Sort</p>
IsDeleted	boolean	<p>System field: Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p> <p>Field Properties: Default on create, Filter, Group, Sort</p>
IsDynamic	boolean	<p><code>true</code> if the channel gets dynamically created on subscribe if necessary, <code>false</code> otherwise.</p> <p>Field Properties: Default on create, Filter, Group, Sort</p>
LastReferencedDate	date	<p>The timestamp for when the current user last viewed a record related to this record.</p> <p>Field Properties: Filter, Sort</p>
LastViewedDate	date	<p>The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p> <p>Field Properties: Filter, Sort</p>
Name	string	<p>Required. Descriptive name of the StreamingChannel. Limit: 80 characters, alphanumeric and <code>"_",""</code> characters only. Must start with <code>/u/</code>. This value identifies the channel and must be unique.</p> <p>Field Properties: Create, Filter, Group, idLookup, Sort, Update</p> <p>Label: Streaming Channel Name</p>
OwnerId	reference	<p>The ID of the owner of the StreamingChannel.</p> <p>Field Properties: Create, Default on create, Filter, Group, Sort, Update</p> <p>Label: Owner Name</p>

TagDefinition

Defines the attributes of child Tag objects.

Supported Calls

`delete(), describesObjects(), query(), retrieve(), search(), undelete(), update()`

Fields

Field	Detail
Name	<p>Type string</p> <p>Properties Filter, Nillable, Update</p> <p>Description Identifies the tag word or phrase.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Nillable, Restricted picklist</p> <p>Description Defines the visibility of a tag. Possible value are:</p> <ul style="list-style-type: none"> • Public: The tag can be viewed and manipulated between all users in an organization. • Personal: The tag can be viewed or manipulated only by a user with a matching OwnerId.

Usage

When you create a tag for a record, an association is created with a corresponding TagDefinition:

- If the value in the tag's Name field is new, a new TagDefinition record is automatically created and becomes the parent of the tag.
- If the value in the tag's Name field already exists in a TagDefinition, that TagDefinition automatically becomes the parent of the tag.

Each TagDefinition record has a one-to-many relationship with its child tag records.

The following standard objects represent tags for records:

- AccountTag
- AssetTag
- CampaignTag
- CaseTag
- ContactTag

- ContractTag
- DocumentTag
- EventTag
- LeadTag
- NoteTag
- OpportunityTag
- SolutionTag
- TaskTag

Custom objects may also be tagged. Tags for custom objects are identified by a suffix of two underscores immediately followed by the word `tag`. For example, a custom object named `Meeting` has a corresponding tag named `Meeting__tag` in that organization's WSDL. `Meeting__tag` is only valid for `Meeting` objects.

`TagDefinition` is useful for mass operations on any tag record. For instance, if you want to rename existing tags, you can search for the appropriate `TagDefinition` object, update it, and the child tag's `Name` values are also changed. The following Java example replaces all `WC` tags with the phrase `West Coast`:

```
public void tagDefinitionSample() {
    String soqlQuery = "SELECT Id, Name FROM TagDefinition " +
        "WHERE Name = 'WC'";
    QueryResult qResult = null;
    try {
        qResult = connection.query(soqlQuery);
        TagDefinition tagDef = (TagDefinition) qResult.getRecords()[0];
        tagDef.setName("West Coast");
        connection.update(new SObject[] {tagDef});
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

When a tag is deleted, its parent `TagDefinition` will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a `TagDefinition` sends it to the Recycle Bin, along with any associated tag entries.

Task

Represents a business activity such as making a phone call or other to-do items. In the user interface, `Task` and `Event` records are collectively referred to as activities.

 **Note:** Task fields related to calls are exclusive to Salesforce CRM Call Center. Additionally, `query()`, `delete()`, and `update()` aren't allowed with tasks related to more than one contact in API versions 23.0 and earlier.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Field Type
AccountId	<p>Type reference</p> <p>Properties Group, Sort, Filter, Nillable</p> <p>Description Represents the ID of the related Account. The AccountId is determined as follows. If the value of WhatId is any of the following objects, then Salesforce uses that object's AccountId.</p> <ul style="list-style-type: none"> • Account • Opportunity • Contract • Custom object that is a child of Account <p>If the value of the WhatId field is any other object, and the value of the WhoId field is a Contact object, then Salesforce uses that contact's AccountId. (If your organization uses Shared Activities, then Salesforce uses the AccountId of the primary contact.)</p> <p>Otherwise, Salesforce sets the value of the AccountId field to null.</p> <p>For information on IDs, see ID Field Type.</p>
ActivityDate	<p>Type date</p> <p>Properties Group, Sort, Create, Filter, Nillable, Update</p> <p>Description Represents the due date of the task. This field has a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. The timestamp is not relevant; do not attempt to alter it in order to accommodate time zone differences. Label is Due Date.</p> <p> Note: This field can't be set or updated for a recurring task (IsRecurrence is true).</p>
CallDisposition	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Represents the result of a given call, for example, "we'll call back," or "call unsuccessful." Limit is 255 characters.</p> <p>Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.</p>

Field	Field Type
CallDurationInSeconds	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Duration of the call in seconds. Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.</p>
CallObject	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Name of a call center. Limit is 255 characters. Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.</p>
CallType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The type of call being answered: Inbound, Internal, or Outbound.</p>
ConnectionReceivedId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that shared this record with your organization. This field is only available if you have enabled Salesforce to Salesforce.</p>
ConnectionSentId	<p>Type reference</p> <p>Properties Filter, Nillable</p> <p>Description ID of the PartnerNetworkConnection that you shared this record with. This field is only available if you have enabled Salesforce to Salesforce. Beginning with API version 15.0, the ConnectionSentId field is no longer supported. The ConnectionSentId field</p>

Field	Field Type
	<p>is still visible, but the value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Contains a text description of the task.</p>
IsArchived	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the event has been archived.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the task has been completed (<code>true</code>) or not (<code>false</code>). Is only set indirectly via the <code>Status</code> picklist. Label is Closed.</p>
IsHighPriority	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates a high-priority task. This field is derived from the <code>Priority</code> field.</p>
IsRecurrence	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the task is scheduled to repeat itself (<code>true</code>) or only occurs once (<code>false</code>). This is a read-only field on update, but not on create. If this field value is <code>true</code>, then <code>RecurrenceStartDateOnly</code>, <code>RecurrenceEndDateOnly</code>, <code>RecurrenceType</code>, and any recurrence fields associated with the given recurrence type must be populated. See Recurring Tasks.</p>

Field	Field Type
IsReminderSet	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether a popup reminder has been set for the task (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInSelfService	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Update</p> <p>Description Indicates whether a task associated with an object can be viewed in the Customer Portal (<code>true</code>) or not (<code>false</code>). If your organization has enabled Communities, tasks marked <code>IsVisibleInSelfService</code> are visible to any external user in the community, as long as the user has access to the record the task was created on. This field is available when Customer Portal or partner portal are enabled OR Communities is enabled and you have Customer Portal or partner portal licenses.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the User who owns the record. Label is Assigned To ID.</p>
Priority	<p>Type picklist</p> <p>Properties Defaulted on create, Group, Sort, Create, Filter, Update</p> <p>Description Required. Indicates the importance or urgency of a task, such as high or low.</p>
RecurrenceActivityId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Read-only. Not required on create. ID of the main record of the recurring task. Subsequent occurrences have the same value in this field.</p>

Field	Field Type
RecurrenceDayOfMonth	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The day of the month in which the task repeats.</p>
RecurrenceDayOfWeekMask	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The day or days of the week on which the task repeats. This field contains a bitmask. The values are as follows:</p> <ul style="list-style-type: none"> • Sunday = 1 • Monday = 2 • Tuesday = 4 • Wednesday = 8 • Thursday = 16 • Friday = 32 • Saturday = 64 <p>Multiple days are represented as the sum of their numerical values. For example, Tuesday and Thursday = $4 + 16 = 20$.</p>
RecurrenceEndDateOnly	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The last date on which the task repeats. This field has a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. The timestamp is not relevant; do not attempt to alter it in order to accommodate time zone differences.</p>
RecurrenceInstance	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The frequency of the recurring task. For example, "2nd" or "3rd."</p>

Field	Field Type
RecurrenceInterval	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The interval between recurring tasks.</p>
RecurrenceMonthOfYear	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The month of the year in which the task repeats.</p>
RecurrenceRegeneratedType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Represents what triggers a repeating task to repeat. Add this field to a page layout together with the RecurrenceInterval field, which determines the number of days between the triggering date (due date or close date) and the due date of the next repeating task in the series. Label is Repeat This Task. This field has the following picklist values:</p> <ul style="list-style-type: none"> • None: The task doesn't repeat. • After due date: The next repeating task will be due the specified number of days after the current task's due date. • After the task is closed: The next repeating task will be due the specified number of days after the current task is closed. • (Task closed): This task, now closed, was opened as part of a repeating series.
RecurrenceStartDateOnly	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The date when the recurring task begins. Must be a date and time before RecurrenceEndDateOnly.</p>
RecurrenceTimeZoneSidKey	<p>Type picklist</p>

Field	Field Type
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The time zone associated with the recurring task. For example, "UTC-8:00" for Pacific Standard Time.</p>
RecurrenceType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Indicates how often the task repeats. For example, daily, weekly, or every nth month (where "nth" is defined in RecurrenceInstance).</p>
ReminderDateTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Represents the time when the reminder is scheduled to fire, if IsReminderSet is set to true. If IsReminderSet is set to false, then the user may have deselected the reminder checkbox in the Salesforce user interface, or the reminder has already fired at the time indicated by the value.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Required. The current status of the task, such as In Progress or Completed. Each predefined Status field implies a value for the IsClosed flag. To obtain picklist values, query the TaskStatus object.</p> <p> Note: This field can't be updated for recurring tasks (IsRecurrence is true).</p>
Subject	<p>Type combobox</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The subject line of the task, such as "Call" or "Send Quote." Limit: 255 characters.</p>

Field	Field Type
TaskSubtype	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Provides standard subtypes to facilitate creating and searching for specific task subtypes. This field isn't updateable.</p> <p>TaskSubtype values:</p> <ul style="list-style-type: none"> • Task • Email • Call
TaskWhoIds	<p>Type JunctionIdList</p> <p>Properties Create, Update</p> <p>Description A string array of contact or lead IDs related to this task. This JunctionIdList field is linked to the TaskWhoRelations child relationship. TaskWhoIds is only available when the shared activities setting is enabled. The first contact or lead ID in the list becomes the primary WhoId if you don't specify a primary WhoId.</p> <p> Warning: Adding a JunctionIdList field name to the fieldsToNull property deletes all related junction records. This action can't be undone.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Update</p> <p>Description The type of task, such as Call or Meeting.</p>
WhatCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Available to organizations that have Shared Activities enabled. Count of related TaskRelations pertaining to WhatId. Count of the WhatId must be 1 or less.</p>
WhatId	<p>Type reference</p>

Field	Field Type
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The <code>WhatId</code> represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. <code>WhatIds</code> are polymorphic. Polymorphic means a <code>WhatId</code> is equivalent to the ID of a related object. The label is <code>Related To ID</code>.</p>
WhoCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Available to organizations that have Shared Activities enabled. Count of related TaskRelations pertaining to <code>WhoId</code>.</p>
WhoId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The <code>Whold</code> represents a human such as a lead or a contact. Wholds are polymorphic. Polymorphic means a <code>Whold</code> is equivalent to a contact's ID or a lead's ID. The label is <code>Name ID</code>. If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the <code>Whold</code> field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is <code>Name ID</code>.</p>

Usage

Recurring Tasks

- Recurring tasks are available in API version 16.0 and later.
- After a task is created, it can't be changed from recurring to nonrecurring or vice versa.
- When you delete a recurring task series through the API, all open and closed task occurrences in the series are removed. However, when you delete a recurring task series through the user interface, only open tasks occurrences (`IsClosed` is `false`) in the series are removed.
- If `IsRecurrence` is `true`, then `RecurrenceStartDateOnly`, `RecurrenceEndDateOnly`, `RecurrenceType`, and any properties associated with the given recurrence type (see the following table) must be populated.
- When you change the `RecurrenceStartDateOnly` field or the recurrence pattern, all open tasks occurrences in the series are deleted and new open task occurrences are created based on the new recurrence pattern. The recurrence pattern is determined by the following fields: `RecurrenceType`, `RecurrenceTimeZoneSidKey`, `RecurrenceInterval`,

`RecurrenceDayOfWeekMask`, `RecurrenceDayOfMonth`, `RecurrenceInstance`, and `RecurrenceMonthOfYear`.

- When you change the value of `RecurrenceEndDateOnly` to an earlier date (for example, from January 20th to January 10th), all open task occurrences in the series with the `ActivityDate` value greater than the new end date value are deleted. Other open and closed task occurrences in the series are not affected.
- When you change the value of `RecurrenceEndDateOnly` to a later date (for example, from January 10th to January 20th), new task occurrences are created up to the new end date. Existing open and closed tasks in the series are not affected.

The following table describes the usage of recurrence fields. Each recurrence type must have all of its properties set. All unused properties must be set to null.

RecurrenceType Value	Properties	Example Pattern
RecurseDaily	<code>RecurrenceInterval</code>	Every second day
RecurseEveryWeekday	<code>RecurrenceDayOfWeekMask</code>	Every weekday - can't be Saturday or Sunday
RecurseMonthly	<code>RecurrenceDayOfMonth</code> <code>RecurrenceInterval</code>	Every second month, on the third day of the month
RecurseMonthlyNth	<code>RecurrenceInterval</code> <code>RecurrenceInstance</code> <code>RecurrenceDayOfWeekMask</code>	Every second month, on the last Friday of the month
RecurseWeekly	<code>RecurrenceInterval</code> <code>RecurrenceDayOfWeekMask</code>	Every three weeks on Wednesday and Friday
RecurseYearly	<code>RecurrenceDayOfMonth</code> <code>RecurrenceMonthOfYear</code>	Every March on the twenty-sixth day of the month
RecurseYearlyNth	<code>RecurrenceDayOfWeekMask</code> <code>RecurrenceInstance</code> <code>RecurrenceMonthOfYear</code>	The first Saturday in every October

JunctionIdList

The `JunctionIdList` field is now implemented in the Event and Task objects. With a single API call, it's easy to create many-to-many relationships between the Event or Task object with contacts, leads, or users.

To create a Task with related Contacts without `JunctionIdList`, you first have to create the task, then use the returned task ID to create the `TaskRelation` records. If the `TaskRelation` save call fails, error handling is your responsibility because the task has already been committed to the database.

```
public void createTasksOld(Contact[] contacts) {
    Task task = new Task();
    task.setSubject("New Task");
    SaveResult[] results = null;
    try {
        results = connection.create(new Task[] {
            task
        });
        if (results[0].isSuccess()) {
            TaskRelation[] relations = new TaskRelation[contacts.size()];
            for (int i = 0; i < contacts.length; i++) {
                relations[i] = new TaskRelation();
                relations[i].setTaskId(results[0].getID());
                relations[i].setContactId(contacts[i].getID());
            }
            connection.create(relations);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

        relations[i].setRelationId(contacts[i].getID());
    }
    results = connection.create(relations);
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

To create a task using `JunctionIdList`, IDs are pulled from the related contacts and both the task and the `TaskRelation` records are created in one API call. If the `TaskRelation` fails, the task is rolled back because it's all done in a single API call.

```

public void createTaskNew(Contact[] contacts) {
    String[] contactIds = new String[contacts.size()];
    for (int i = 0; i < contacts.size(); i++) {
        contactIds[i] = contacts[i].getID();
    }
    Task task = new Task();
    task.setSubject("New Task");
    task.setTaskWhoIds(contactIds);
    SaveResult[] results = null;
    try {
        results = connection.create(new Task[] {
            task
        });
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
}

```

SEE ALSO:

[Object Basics](#)

TaskFeed

Represents a single feed item in the feed on a Task. This object is available in API version 20.0 and later.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Moderate Chatter”

 **Note:** Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of TaskFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p>

Field	Details
	<p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p>

Field	Details
	<p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none">• <code><p></code>

Field	Details
	<p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none">• • <code>• <i>• <u>• <s>• • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>

Field	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the task record. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> AdvancedTextPost—created when a user posts a group announcement. AnnouncementPost—Not used. ApprovalPost—generated when a user submits an approval. BasicTemplateFeedItem—Not used. CanvasPost—a post made by a canvas app posted on a feed. CollaborationGroupCreated—generated when a user creates a public group. CollaborationGroupUnarchived—Not used. ContentPost—a post with an attached file. CreatedRecordEvent—generated when a user creates a record from the publisher. DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. LinkPost—a post with an attached URL. PollPost—a poll posted on a feed. ProfileSkillPost—generated when a skill is added to a user's Chatter profile. QuestionPost—generated when a user posts a question. ReplyPost—generated when Chatter Answers posts a reply. RipplePost—generated when a user creates a Thanks badge in Work.com.

Field	Details
	<ul style="list-style-type: none">• <code>TextPost</code>—a direct text entry on a feed.• <code>TrackedChange</code>—a change or group of changes to a tracked field.• <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to <code>CaseFeed</code>:</p> <ul style="list-style-type: none">• <code>CaseCommentPost</code>—generated event when a user adds a case comment for a case object• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case
	<p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a task record.

TaskPriority

Represents the importance or urgency of a Task, such as High, Normal, or Low.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
<code>ApiName</code>	Type string Properties Filter, Group, NillableSort Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.
<code>IsDefault</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether this is the default task priority value (<code>true</code>) or not (<code>false</code>) in the picklist. Only one value in the picklist can be the default value.
<code>IsHighPriority</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether this task priority value represents a high priority Task (<code>true</code>) or not (<code>false</code>). Multiple task priority values can represent a high-priority Task.
<code>MasterLabel</code>	Type string Properties Filter, Group, Nillable, Sort

Field	Details
	<p>Description Master label for this task priority value. This display value is the internal label that does not get translated. Limit: 255 characters.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Nillable, Group, Sort</p> <p>Description Number used to sort this value in the task priority picklist. These numbers are not guaranteed to be sequential, as some previous task priority values might have been deleted.</p>

Usage

This object represents a value in the task priority picklist. The task priority picklist provides additional information about the importance of a Task, such as whether a given priority value represents a high priority. Your client application can query on this object to retrieve the set of values in the task priority picklist, and then use that information while processing Task objects to determine more information about a given task. For example, the application could test whether a given Task is high priority based on its `Priority` value and the value of the `IsHighPriority` in the associated TaskPriority object.

SEE ALSO:

[Object Basics](#)

TaskRelation

Represents the relationship between a task and a lead, contacts, and other objects related to the task. If Shared Activities is enabled, this object doesn't support triggers, workflow, or data validation rules. This object is available in API version 24.0 and later.

TaskRelation is only available if you've enabled Shared Activities in your organization.

TaskRelation allows the following relationships:

- A task can be related to one lead or up to 50 contacts.
- A task can also be related to one account, asset, campaign, case, contract, opportunity, product, solution, or custom object.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `queryAll()`, `retrieve()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the Account ID of the relation. For information on IDs, see ID Field Type.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a task has been deleted; label is Deleted. When a TaskRelation record is deleted, it isn't moved to the Recycle Bin and can't be undeleted, unless the record was cascade-deleted when the parent object was deleted. Don't use the <code>IsDeleted</code> field to detect deleted records in SOQL queries or <code>queryAll()</code> calls on directly deleted relation records. Instead, use the call <code>getDeleted()</code>.</p>
IsWhat	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the relation is an Account, Opportunity, Campaign, Case, other standard object, or a custom object. Value is <code>false</code> if <code>RelationId</code> is a contact or lead and <code>true</code> otherwise.</p>
RelationId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Indicates the <code>WhatId</code> or <code>WhoId</code> in the relationship. For more information, see Task. For information on IDs, see ID Field Type.</p>
TaskId	<p>Type reference</p>

Field Name	Details
	Properties Create, Filter, Group, Sort
	Description Represents the ID of the associated Task. For information on IDs, see ID Field Type .

Usage

See contacts associated with a task

```
public void queryWhosOfTaskSample() {  
    String soqlQuery = "SELECT Id, Subject, (SELECT RelationId, Relation.Name, IsWhat  
    from TaskRelations WHERE IsWhat = false) FROM Task WHERE Id = '00T x0000005OKEN''";  
    QueryResult qResult = null;  
    try {  
        qResult = connection.query(soqlQuery);  
        TaskRelation relation1 =  
(TaskRelation)qResult.getRecords()[0].getTaskRelations().getRecords()[0];  
    }catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
}
```

SEE ALSO:

[Task](#)
[TaskWhoRelation](#)

TaskStatus

Represents the status of a Task, such as Not Started, Completed, or Closed.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
ApiName	<p>Type string</p> <p>Properties Filter, Group, NillableSort</p> <p>Description Uniquely identifies a picklist value so it can be retrieved without using an id or master label.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this task status value represents a closed Task (<code>true</code>) or not (<code>false</code>). Multiple task status values can represent a closed Task.</p>
IsDefault	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether this is the default task status value (<code>true</code>) or not (<code>false</code>) in the picklist.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Master label for this task status value. This display value is the internal label that does not get translated. Limit: 255 characters.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Number used to sort this value in the task status picklist. These numbers are not guaranteed to be sequential, as some previous task status values might have been deleted.</p>

Usage

This object represents a value in the task status picklist. The task status picklist provides additional information about the status of a Task, such as whether a given status value represents an open or closed task. Your client application can query this object to retrieve the set of values in the task status picklist, and then use that information while processing Task records to determine more information about a given task. For example, the application could test whether a given task is open or closed based on the Task Status value and the value of the IsClosed property in the associated TaskStatus record.

SEE ALSO:

[Object Basics](#)

TaskTag

Associates a word or short phrase with a Task.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ItemId	Type reference Properties Create, Filter Description ID of the tagged item.
Name	Type string Properties Create, Filter Description Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.
TagDefinitionId	Type reference Properties Filter

Field Name	Details
	<p>Description ID of the parent TagDefinition object that owns the tag.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist</p> <p>Description Defines the visibility of a tag.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • Public—The tag can be viewed and manipulated by all users in an organization. • Personal—The tag can be viewed or manipulated only by a user with a matching OwnerId.

Usage

TaskTag stores the relationship between its parent TagDefinition and the Task being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

TaskWhoRelation

Represents the relationship between a task and a lead or contacts. This object is available in API version 29.0 and later.

This derived object is a filtered version of the [TaskRelation](#) on page 1821 object; that is, **IsParent** is *true* and **IsWhat** is *false*. It doesn't represent relationships to accounts, opportunities, or other objects.

TaskWhoRelation allows a variable number of relationships: one lead or up to 50 contacts. Available only if you've enabled Shared Activities for your organization.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
RelationId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the contacts or lead related to the task.</p>
TaskId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the task.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates whether the person related to the task is a lead or contact.</p>

Usage

Apex example that queries contacts associated with a task

```
public void queryWhosOfTaskSample() {
    String soqlQuery = "SELECT Id, Subject, (SELECT RelationId, Relation.Name, IsWhat
from TaskWhoRelations) FROM Task WHERE Id = '00Tx0000005OKEN'";
    QueryResult qResult = null;
    try {
        qResult = connection.query(soqlQuery);
        TaskWhoRelation relation1 =
        (TaskWhoRelation)qResult.getRecords()[0].getTaskWhoRelations().getRecords()[0];
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

SEE ALSO:

[Task](#)
[TaskRelation](#)

TenantSecret

This object stores an encrypted organization-specific key fragment that is used with the master secret to produce organization-specific data encryption keys. This object is available in API version 34.0 and later.

You can rotate the tenant secret once every four hours in a sandbox organization or every 24 hours in production organizations.

 **Note:** This information is about Shield Platform Encryption and not Classic Encryption.

Supported Calls

`create()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
Description	<p>Type string</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the tenant secret.</p>
SecretValue	<p>Type base64</p> <p>Properties Nillable, Update</p> <p>Description The encrypted 256-bit secret value encoded in base64.</p>
Status	<p>Type Restricted picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The current status of the tenant secret. Values are:</p> <p>ACTIVE Can be used to encrypt and decrypt new or existing data.</p> <p>ARCHIVED Cannot encrypt new data. Can be used to decrypt data previously encrypted with this key when it was active.</p>

Field Name	Details
	<p>DESTROYED</p> <p>Cannot encrypt or decrypt data. Data encrypted with this key when it was active can no longer be decrypted. Files and attachments encrypted with this key can no longer be downloaded.</p>
Version	<p>Type int</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The version number of this secret. The version number is unique within your organization.</p>

Usage

Use this object to create or update your organization-specific tenant secret. For example, you can build an automated tenant secret creation and activation solution similar to the following.

1. Start by creating an Apex class to create the new tenant secret.

```
global class CreateNewSecret implements Schedulable {
    global void execute(SchedulableContext SC) {
        TenantSecret k = new TenantSecret ();
        k.description = 'Created new secret from scheduled job';
        insert k;
    }
}
```

2. Schedule the Apex class to run at the specified interval.

This Apex code only needs to be run a single time to schedule the job. This code runs the job every 90 days.

```
CreateNewSecret secret = new CreateNewSecret();
String schedule = '0 0 0 1 JAN,APR,JUL,OCT ?';
String jobID = system.schedule('Automated secret creation and activation', schedule,
secret);
```

3. Validate that the job is scheduled.

4. Validate that tenant secrets are created after the job is run.

Here is an example of destroying a tenant secret.

 **Warning:** Your tenant secret is unique to your organization and to the specific data to which it applies. Once you destroy a tenant secret, related data is not accessible unless you previously exported the key and then import the key back into Salesforce.

```
TenantSecret secret = [SELECT Id FROM TenantSecret WHERE Version = 2];
secret.SecretValue = NULL;
update secret;
```

Territory

Represents a flexible collection of accounts and users where the users have at least read access to the accounts, regardless of who owns the accounts. Only available if territory management has been enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Fields

Field	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Account access level granted to users assigned to this territory.</p>
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Nillable, Group, Sort, Update</p> <p>Description Case access level granted to users assigned to this territory.</p>
ContactAccessLevel	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A value that represents the type of access granted to the target Group, UserRole, or User for any associated contacts. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p> Note: When DefaultContactAccess is set to "Controlled by Parent," you can't create or update this field.</p>
Description	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Nullable, Sort, Update</p>
	<p>Description A description of the territory that is 1,000 characters or less.</p>
DeveloperName	<p>Type string</p>
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>
	<p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Territory Name in the user interface.</p>
	<p>This field is available in API version 24.0 and later.</p>
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
ForecastUserId	<p>Type reference</p>
	<p>Properties Create, Filter, Group, Nullable, Sort, Update</p>
	<p>Description ID of the Forecast Manager, who is the user to whom forecasts from this territory's child territories roll up.</p>
MayForecastManagerShare	<p>Type boolean</p>
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>
	<p>Description Indicates whether the forecast manager can manually share their own forecast.</p>
Name	<p>Type string</p>
	<p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description A name for the territory. Limit is 80 characters. Corresponds to Label on the user interface.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Opportunity access level granted to users assigned to this territory.</p>
ParentTerritoryID	<p>Type reference</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Territory immediately above this territory in the territory hierarchy. Label is Parent Territory ID.</p>
RestrictOppTransfer	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Update</p> <p>Description Indicates whether the opportunities associated with this territory are kept within the bounds of this territory and this territory's children when account assignment rules are run (<code>true</code>), or if opportunities associated with this territory can be assigned to other nodes of the territory hierarchy when account assignment rules are run (<code>false</code>). Label is Confine Opportunity Assignment.</p>

Usage

Use the Territory object to query your organization's territory hierarchy. Use it to obtain valid territory IDs when querying or modifying records associated with territories.

SEE ALSO:

- [AccountTerritoryAssignmentRule](#)
- [AccountTerritoryAssignmentRuleItem](#)
- [UserTerritory](#)

Territory2

Represents a sales territory. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`create(), delete(), describeLayout(), describeObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your organization's sharing settings. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Represents the default account record access levels for users that are assigned to the territory. Values are:</p> <ul style="list-style-type: none">• Read Only• Read/Write• Owner
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Represents the default case record access levels for users that are assigned to the territory. Values are:</p> <ul style="list-style-type: none">• Private• Read Only• Read/Write
ContactAccessLevel	<p>Type picklist</p>

Field Name	Details
	<p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Represents the default contact record access levels for users that are assigned to the territory. Values are:</p> <ul style="list-style-type: none"> • Private • Read Only • Read/Write
Description	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The description of the territory.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. The field label in the user interface is Territory Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the territory. The field label in the user interface is Label.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description Represents the default opportunity record access levels for users that are assigned to the territory. Values are:</p> <ul style="list-style-type: none"> • Private • Read Only • Read/Write
ParentTerritory2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID of the territory's parent territory (if any). If the territory has no parent territory, this value is <code>null</code>.</p>
Territory2ModelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the territory model that the territory belongs to.</p>
Territory2TypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the territory type that the territory belongs to.</p>

Territory2Model

Represents a territory model. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your organization's sharing settings. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
ActivatedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the territory model was activated.</p>
DeactivatedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the territory model was archived.</p>
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The description of the territory model.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. The field label in the user interface is <code>Territory Model Name</code>.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>

Field Name	Details
LastOppTerrAssignEndDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read-only. The date when the opportunity territory assignment filter was last run. Used for Filter-Based Opportunity Territory Assignment (Pilot in Spring '15 / API version 33).</p>
LastRunRulesEndDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the last rules run was completed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The territory model name. The field label in the user interface is <code>Label</code>.</p>
State	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The state of the territory model. Values are: <code>Planning</code>, <code>Activating</code>, <code>Activation Failed</code>, <code>Active</code>, <code>Archiving</code>, <code>Archiving Failed</code>, <code>Archived</code>, <code>Deleting</code>, and <code>Deletion Failed</code>.</p>

Territory2ModelHistory

Represents the history of changes to the values in the fields on a territory model. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field whose value was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the changed field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The previous value of the changed field.</p>
Territory2ModelId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the territory model whose history is tracked.</p>

Usage

This object is automatically generated whenever any field value changes on a territory model record. Use this object it to identify those changes.

Territory2Type

Represents a category for territories (Territory2). Every Territory2 must have a Territory2Type. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

All users (including standard users) have access to this object in the user interface

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The description of the territory type.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. The field label in the user interface is Territory Type Name.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the master label in the user interface.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description Required The user interface label for the territory type.</p>
Priority	<p>Type int</p> <p>Properties Create, Filter, Group, SortUpdate</p> <p>Description Required. Used for Filter-Based Opportunity Territory Assignment (Pilot in Spring '15 / API version 33). Lets you specify a priority for a territory type. For opportunity assignments, the filter examines all territories assigned to the account that the opportunity is assigned to. The account-assigned territory whose territory type priority is highest is then assigned to the opportunity. The <code>priority</code> field value on each territory type must be unique. Further, if there are multiple territories with the same territory type (and therefore the same priority) assigned to the account, no territory is not assigned to the opportunity.</p>

TestSuiteMembership

Associates an Apex class with an ApexTestSuite. This object is available in API version 36.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Description
ApexClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The Apex class whose tests are to be executed.</p>
ApexTestSuiteId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Description
	Description
	The test suite to which the Apex class is assigned.

Usage

Insert a TestSuiteMembership object using an API call to associate an Apex class with an ApexTestSuite object. (ApexTestSuite and TestSuiteMembership aren't editable through Apex DML.) To remove the class from the test suite, delete the TestSuiteMembership object. If you delete an Apex test class or test suite, all TestSuiteMembership objects that contain that class or suite are deleted.

The following SOQL query returns the membership object that relates this Apex class to this test suite.

```
SELECT Id FROM TestSuiteMembership WHERE ApexClassId = '01pD0000000Fhy9IAC'
    AND ApexTestSuiteId = '05FD00000004CDBMA2'
```

SEE ALSO:

[ApexTestSuite](#)

ThirdPartyAccountLink

Represents the list of external users who authenticated using an Auth. Provider. This object is available in API version 32.0 and later.

A list of third-party account links is generated when users of an organization authenticate using an external Auth. Provider. Use this object to list and revoke a given user's social sign-on connections (such as Facebook[®]).

Supported Calls

`describeSObjects()`, `query()`

Fields

Field Name	Details
Handle	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The username in the third-party system.</p>
IsNotSsoUsable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	Description
	<p>Support for single sign-on.</p>
	<p>If <code>true</code>, the link cannot be used for a single sign-on flow. It is only available OAuth access and refresh tokens.</p>
Provider	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The third-party account provider name.</p>
RemoteIdentifier	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The unique ID for the user in the third-party system.</p>
SsoProvider	Type
	AuthProvider on page 347
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The foreign key to the AuthProvider of the third-party system.</p>
SsoProviderId	Type
	reference
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The ID associated with the <code>SsoProvider</code> value.</p>
SsoProviderName	Type
	string
	Properties
	Filter, Group, Nillable, Sort
	Description
	<p>The name associated with the AuthProvider of the third-party system, in case the user has no access to the provider foreign key (the <code>SsoProvider</code> value).</p>

Field Name	Details
ThirdPartyAccountLinkKey	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A concatenated string including the organization ID, the <code>SsoProviderId</code> value, the <code>SsoProvider</code> value, and the <code>RemoteIdentifier</code> value.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The Salesforce user associated with this third-party account link.</p>

Usage

Use this object makes to build custom screens to manage a user's third-party account links.

Administrators (with the "Manage Users" permission) querying this object can see all the links for all users in the organization. Without the "Manage Users" permission, users can only retrieve thier own links. A user may not have access to the `SsoProvider` value (the foreign key). In this case, use the `SsoProviderName` to render the name of the provider for the associated link.

Use the Apex method `Auth.AuthToken.revokeAccess()` to revoke a link.

In API version 34.0 and later, this object was enhanced to help manage high instance counts. A `query()` call returns up to 500 rows. A `queryMore()` call returns 500 more, up to 2500 total. No more records are returned after 2500. To make sure you don't miss any records, issue a `COUNT()` query in a `SELECT` clause for `ThirdPartyAccountLink`. This gives you the total number of records. If there are more than 2500 records, divide your query by filtering on fields, like `UserId`, to return subsets of less than 2500 records.

TimeSlot

Represents a period of time on a specified day of the week during which field service work can be performed. Operating hours consist of one or more time slots. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
DayOfWeek	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The day of the week when the time slot takes place.</p>
EndTime	<p>Type time</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The time when the time slot ends.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.</p>
OperatingHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The operating hours that the time slot belongs to. An operating hours' time slots appear in the Operating Hours related list.</p>

Field Name	Details
StartTime	<p>Type time</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The time when the time slot starts.</p>
TimeSlotNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the time slot. The name is auto-populated to a day and time format—for example, <code>Monday 9:00 AM – 10:00 PM</code>—but you can manually update it if you wish.</p>
Type	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of time slot. Possible values are <code>Normal</code> and <code>Extended</code>. You may choose to use <code>Extended</code> to represent overtime shifts.</p>

Usage

Operating hours are comprised of time slots, which indicate the hours of operation for a particular day. After you create operating hours, create time slots for each day in the Time Slots related list. For example, if the operating hours should be 8 AM to 5 PM Monday through Friday, create five time slots, one per day. To reflect breaks such as lunch hours, create multiple time slots in a day: for example, `Monday 8:00 AM – 12:00 PM` and `Monday 1:00 PM – 5:00 PM`.



Tip: Time slots don't come with any built-in rules, but you can create Apex triggers that limit time slot settings in your org. For example, you may want to restrict the start and end times on time slots to half-hour increments, or to prohibit end times later than 8 PM.

TimeSlotHistory

Represents the history of changes made to tracked fields on a time slot. This object is available in API version 38.0 and later.

Supported Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

Field Service Lightning must be enabled in your organization, and field tracking for time slot fields must be configured.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
TimeSlotId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the time slot being tracked. The history is displayed on the detail page for this record.</p>

Topic

Represents a topic on a Chatter post or record. This object is available in API version 28.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), search(), update(), upsert()`

Fields

Field Name	Details
Description	<p>Type string</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Description of the topic.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p> Note: You can change only the spacing and capitalization of a topic name with the update property.</p> <p>Description Name of the topic.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description Identifier of the Community to which the Topic belongs. This field is available only if Salesforce Communities is enabled in your organization.</p>
TalkingAbout	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description Number of people talking about the topic over the last two months, based on factors such as topic additions and comments on posts with the topic.</p>

Usage

Use this object to query a specific topic or to get a list of all topics, even those used solely in private groups and on records, and the number of people talking about them.

Use this object to create, edit, or delete topics. To create a topic, you must have the "Create Topics" permission. To edit a topic, you must have the "Edit Topics" permission. To delete a topic, you must have the "Delete Topics" or "Modify All Data" permission.

TopicAssignment

Represents the assignment of a topic to a specific feed item, record, or file. This object is available in API version 28.0 and later.

Administrators must enable topics for objects before users can add topics to records of that object type. Topics for most objects are available in API version 30.0 and later. Topics for ContentDocument are available in API version 37.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
EntityId	Type reference Properties Create, Filter, Group, Sort Description Identifier of the feed item, record, or file.
EntityKeyPrefix	Type string Properties Filter, Group, idLookup, Sort Description The first three digits of the EntityID field, which identify the object type (account, opportunity, etc). This read-only field is available in API version 32.0 and later. Interface label is "Record Key Prefix," which appears only in reports.
EntityType	Type string Properties Filter, Group, Nillable, Sort

Field Name	Details
	<p>Description The standard name for the object type (account, opportunity, etc). This read-only field is available in API version 33.0 and later. Interface label is "Object Type," which appears only in reports.</p> <p> Tip: In most cases, you should use this field rather than <code>EntityKeyPrefix</code>, which exists primarily to support older reports.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description Identifier of the community to which the TopicAssignment belongs. This field is available only if Salesforce Communities is enabled in your organization.</p>
TopicId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Identifier of the topic.</p>

Usage

Use this object to query the assignments of topics to feed items, records, or files. To assign or remove topics, you must have the "Assign Topics" permission.

In SOQL `SELECT` syntax, this object supports nested semi-joins, allowing queries on Knowledge articles assigned to specific topics. For example:

```
SELECT parentId FROM KnowledgeArticleViewStat
  WHERE parentId in (SELECT KnowledgeArticleId FROM KnowledgeArticleVersion
    WHERE publishStatus = 'Online' AND language = 'en_US'
    AND Id in (select EntityId from TopicAssignment where TopicId ='0T0xx0000000xxx'))
```

No SOQL limit if logged-in user has "View All Data" permission. If not, do one of the following:

- Specify a `LIMIT` clause of 1,100 records or fewer.
- Filter on `Id` or `Entity` when using a `WHERE` clause with "=".

SEE ALSO:

[Topic](#)
[FeedItem](#)

TopicFeed

Represents a single feed item on a topic page. This object is available in API version 29.0 and later.

Supported Calls

`delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Moderate Chatter"

 **Note:** Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
Description	
	<p>When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator. The <code>ConnectionId</code> contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is the size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>

Field Name	Details
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is the MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code>

Field Name	Details
	<p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p>
	 Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a CollaborationGroup or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks.
ParentId	<p>Type reference</p>

Field Name	Details
	Properties Filter, Group, Sort
	Description ID of the topic.
RelatedRecordId	Type reference
	Properties Group, Nillable, Sort
	Description ID of the ContentVersion object associated with a ContentPost. For Work.com thanks posts, it's the ID of the WorkThanks object associated with a RypplePost. This field is typically null for all posts except ContentPost and RypplePost.
Title	Type string
	Properties Group, Nillable, Sort
	Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.
Type	Type picklist
	Properties Filter, Group, Nillable, Restricted picklist, Sort
	Description The type of FeedItem:
	<ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).
	<p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p>
	<ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval.

Field Name	Details
	<ul style="list-style-type: none">• <code>BasicTemplateFeedItem</code>—Not used.• <code>CanvasPost</code>—a post made by a canvas app posted on a feed.• <code>CollaborationGroupCreated</code>—generated when a user creates a public group.• <code>CollaborationGroupUnarchived</code>—Not used.• <code>ContentPost</code>—a post with an attached file.• <code>CreatedRecordEvent</code>—generated when a user creates a record from the publisher.• <code>DashboardComponentAlert</code>—generated when a dashboard metric or gauge exceeds a user-defined threshold.• <code>DashboardComponentSnapshot</code>—created when a user posts a dashboard snapshot on a feed.• <code>LinkPost</code>—a post with an attached URL.• <code>PollPost</code>—a poll posted on a feed.• <code>ProfileSkillPost</code>—generated when a skill is added to a user's Chatter profile.• <code>QuestionPost</code>—generated when a user posts a question.• <code>ReplyPost</code>—generated when Chatter Answers posts a reply.• <code>RipplePost</code>—generated when a user creates a Thanks badge in Work.com.• <code>TextPost</code>—a direct text entry on a feed.• <code>TrackedChange</code>—a change or group of changes to a tracked field.• <code>UserStatus</code>—automatically generated when a user adds a post. Deprecated.

The following values appear in the `Type` picklist for all feed objects but apply only to [CaseFeed](#):

- `AttachArticleEvent`—generated event when a user attaches an article to a case.
- `CallLogPost`—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.
- `CaseCommentPost`—generated event when a user adds a case comment for a case object.
- `ChangeStatusPost`—generated event when a user changes the status of a case.
- `ChatTranscriptPost`—generated event when Live Agent transcript is saved to a case.
- `EmailMessageEvent`—generated event when an email related to a case object is sent or received.
- `FacebookPost`—generated when a Facebook post is created from a case. Deprecated.

Field Name	Details
	<ul style="list-style-type: none"> • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

TopicLocalization—Beta

Represents the translated version of a topic name. Topic localization applies only to navigational and featured topics in communities. This object is available in API version 33.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`update()`,`upsert()`

Special Access Rules

Users with the Translation Workbench enabled can view topic translations, but the “Customize Application,” “Manage Translation,” or “Manage Categories” permission is required to create or update them.

Fields

Field Name	Details
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The combined language and locale ISO code, which controls the language for labels displayed in an application. (The values in this field are not related to the default locale selection.)</p> <p>This picklist contains the following fully-supported languages:</p> <ul style="list-style-type: none">• Chinese (Simplified): zh_CN• Chinese (Traditional): zh_TW• Danish: da• Dutch: nl_NL• English: en_US• Finnish: fi• French: fr• German: de• Italian: it• Japanese: ja• Korean: ko• Norwegian: no• Portuguese (Brazil): pt_BR• Russian: ru• Spanish: es• Spanish (Mexico): es_MX• Swedish: sv• Thai: th <p>The following end-user only languages are available.</p> <ul style="list-style-type: none">• Arabic: ar• Bulgarian: bg• Croatian: hr• Czech: cs• English (UK): en_GB• Greek: el• Hebrew: iw• Hungarian: hu

Field Name	Details
	<ul style="list-style-type: none">Indonesian: <code>in</code>Polish: <code>pl</code>Portuguese (Portugal): <code>pt_PT</code>Romanian: <code>ro</code>Slovak: <code>sk</code>Slovenian: <code>sl</code>Turkish: <code>tr</code>Ukrainian: <code>uk</code>Vietnamese: <code>vi</code> <p>The following platform languages are available for organizations that use Salesforce exclusively as a platform.</p> <ul style="list-style-type: none">Albanian: <code>sq</code>Arabic (Algeria): <code>ar_DZ</code>Arabic (Bahrain): <code>ar_BH</code>Arabic (Egypt): <code>ar_EG</code>Arabic (Iraq): <code>ar_IQ</code>Arabic (Jordan): <code>ar_JO</code>Arabic (Kuwait): <code>ar_KW</code>Arabic (Lebanon): <code>ar_LB</code>Arabic (Libya): <code>ar LY</code>Arabic (Morocco): <code>ar_MA</code>Arabic (Oman): <code>ar_OM</code>Arabic (Qatar): <code>ar_QA</code>Arabic (Saudi Arabia): <code>ar_SA</code>Arabic (Sudan): <code>ar_SD</code>Arabic (Syria): <code>ar_SY</code>Arabic (Tunisia): <code>ar_TN</code>Arabic (United Arab Emirates): <code>ar_AE</code>Arabic (Yemen): <code>ar_YE</code>Armenian: <code>hy</code>Basque: <code>eu</code>Bosnian: <code>bs</code>Bengali: <code>bn</code>Chinese (Simplified—Singapore): <code>zh_SG</code>Chinese (Traditional—Hong Kong): <code>zh_HK</code>English (Australia): <code>en_AU</code>English (Canada): <code>en_CA</code>English (Hong Kong): <code>en_HK</code>

Field Name	Details
	<ul style="list-style-type: none">English (India): en_INEnglish (Ireland): en_IEEnglish (Malaysia): en_MYEnglish (Philippines): en_PHEnglish (Singapore): en_SGEnglish (South Africa): en_ZAEstonian: etFrench (Belgium): fr_BEFrench (Canada): fr_CAFrench (Luxembourg): fr_LUFrench (Switzerland): fr_CHGeorgian: kaGerman (Austria): de_ATGerman (Luxembourg): de_LUGerman (Switzerland): de_CHHindi: hiIcelandic: isIrish: gaItalian (Switzerland): it_CHLatvian: lvLithuanian: ltLuxembourgish: lbMacedonian: mkMalay: msMaltese: mtRomanian (Moldova): ro_MDMontenegrin: sh_MERomansh: rmSerbian (Cyrillic): srSerbian (Latin): shSpanish (Argentina): es_ARSpanish (Bolivia): es_BOSpanish (Chile): es_CLSpanish (Colombia): es_COSpanish (Costa Rica): es_CRSpanish (Dominican Republic): es_DOSpanish (Ecuador): es_ECSpanish (El Salvador): es_SV

Field Name	Details
	<ul style="list-style-type: none">Spanish (Guatemala): es_GTSpanish (Honduras): es_HNSpanish (Nicaragua): es_NISpanish (Panama): es_PASpanish (Paraguay): es_PYSpanish (Peru): es_PESpanish (Puerto Rico): es_PRSpanish (United States): es_USSpanish (Uruguay): es_UYSpanish (Venezuela): es_VETagalog: tlTamil: taUrdu: urWelsh: cy
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none">In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>ID that identifies the topic. After a TopicLocalization record is created, this ID can't be modified.</p>
Value	Type
	<p>string</p> <p>Properties</p> <p>Create, Filter, Sort, Update</p> <p>Description</p> <p>The translated text for the topic name. Label is Topic Name Translation.</p>

TwoFactorInfo

Stores a user's secret for two-factor operations. Use this object when customizing two-factor authentication in your organization. This object is available in API version 32.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

You need the "Manage Two-Factor Authentication in API" permission to create or update this object.

Fields

Field Name	Details
SharedKey	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Group, Sort, Update</p> <p>Description</p> <p>This field is never read-enabled, though it is write-enabled. A request for this value always returns <code>null</code>. The value must be a base32-encoded string of a 20-byte secret.</p> <p>You can use the Apex method <code>Auth.SessionManagement.getQrCode()</code> to get a value to write to this field.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The two-factor method.</p> <ul style="list-style-type: none"> • TOTP—The time-based one-time password. • HOTP—Reserved for future use.
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID for the user who's associated with the authentication secret.</p>

TwoFactorMethodsInfo

Stores information about which identity verification methods a user has registered. This object is available in API version 37.0 and later.

Supported Calls

`describeSObjects(), query()`

Special Access Rules

You need the "Manage Two-Factor Authentication in API" permission to access this object.

Fields

Field Name	Details
ExternalId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
	<p>Description A unique system-generated numerical identifier for the user.</p>
HasSalesforceAuthenticator	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has connected the Salesforce Authenticator mobile app. The user can verify identity by approving a notification sent to the app. If the user sets a trusted location in the app, Salesforce Authenticator verifies automatically when the user is in the trusted location.</p>
HasTempCode	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has a temporary verification code generated by a Salesforce admin or user with "Manage Two-Factor Authentication in User Interface" permission.</p>
HasTotp	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has connected an authenticator app that generates verification codes, also known as time-based one-time passwords (TOTP). The user can verify identity by entering a code generated by the app.</p>
HasU2F	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has registered a U2F security key. The user can verify identity by inserting the security key into a USB port to generate credentials.</p>
HasVerifiedMobileNumber	<p>Type boolean</p>

Field Name	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the user has verified a mobile phone number. Salesforce can text a verification code to the user at that number.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the user who's associated with the identity verification methods.</p>

TwoFactorTempCode

Stores information about a user's temporary identity verification code. This object is available in API version 37.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

You need the "Manage Two-Factor Authentication in API" permission to access this object.

Fields

Field Name	Details
Expiration	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date and time when the temporary verification code expires. The code expires in 1 to 24 hours after it's generated. Salesforce admins and non-admin users with the "Manage Two-Factor Authentication in User Interface" permission set the expiration time when generating the code.</p>
Identifier	<p>Type string</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The unique identifier for the temporary code. This is a required field that can take any value.</p>
TempCode	<p>Type encryptedstring</p> <p>Description A request for this value always returns null.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID for the user who's associated with the temporary verification code.</p>

UndecidedEventRelation

Represents invitees with the status Not Responded for a given event. This object is available in API versions 29.0 and later.

Supported Calls

`describeSObjects(), query(), retrieve()`

Fields

Field Name	Details
EventId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates the ID of the event.</p>
RelationId	<p>Type reference</p>

Field Name	Details
	Properties Filter, Group, Nillable, Sort
	Description Indicates the ID of the invitee.
RespondedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description This field is always null.
Response	Type string
	Properties Filter, Group, Nillable, Sort
	Description Indicates the content of the response field. Label is Comment.
Type	Type string
	Properties Filter, Group, Nillable, Sort
	Description Indicates whether the invitee is a user, lead or contact, or resource.

Usage

Query invitees who have not responded to an invitation to an event

```
SELECT eventId, type, response FROM UndecidedEventRelation WHERE
eventId='00UTD000000ZH5LA'
```

SEE ALSO:

[AcceptedEventRelation](#)

[DeclinedEventRelation](#)

User

Represents a user in your organization.

Supported Calls

`create(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

- To create or update a User record, you must have the “Manage Internal Users” permission. Additionally, if the user is a Customer Portal user, you must have the “Edit Self-Service Users” permission, and if the user is a partner portal user, you must have the “Manage External Users” permission.
- If [Communities](#) is enabled in your organization, to create or update external users for Customer Portal, partner portal, or Communities, you must also have the “Manage External Users” permission.
- Information in hidden fields in a user’s profile is not searchable by external users (with a portal profile) in a community. For example, if a user in a community has a hidden email address and an external user searches for it, the user record is not returned in the search results. Hidden field values are also not returned when external users perform searches on non-hidden fields. So if an external user searches for a user’s name (cannot be hidden), any hidden field values associated with the user record (for example, a hidden email address) are not returned in the search results.

Internal users from your organization who belong to the same community, however, can both search for and view hidden field values in search results.

- When requested by portal users, queries that look up to the User object, such as `owner.name` or `owner.email` might not return values when the portal user making the request doesn’t have Read access to the User record being queried.

The behavior depends on the number of domains associated with the lookup field. If the object can look up to more than one domain, `owner.name` returns a value, but other detail fields don’t. For example, Case owner can look up to the User or Queue objects. In this case, portal users can see only the value of `owner.name`. Other User detail fields, such as `owner.email` or `owner.phone` don’t return a value.

If the object can look up to only a single domain, such as Account owner, then no detail fields return values, including `owner.name`.

- Changing ownership of a record by updating its `OwnerId` field requires the user making the change to have both the “Transfer Record” permission and Read access to the User record of the new record owner.

Fields

Field	Details
<code>AboutMe</code>	<p>Type textarea</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Information about the user, such as areas of interest or skills. This field is available even if Chatter is disabled.</p>
<code>AccountId</code>	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the Account associated with a Customer Portal user. This field is null for Salesforce users.</p>
Address (beta)	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
Alias	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The user's alias. For example, <code>j smith</code>.</p>
BadgeText	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The text description of a user badge that appears over a user's photo. Users of the same Chatter user type (internal, external) are badged. Different user types are not badged.</p>
BannerPhotoUrl	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the user's banner photo. This field is available in API version 36.0 and later.</p>
CallCenterId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description If Salesforce CRM Call Center is enabled, represents the call center to which this user is assigned.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city associated with the user. Up to 40 characters allowed.</p>
CommunityNickname	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Name used to identify this user in the Community application, which includes the ideas and answers features.</p>
CompanyName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The name of the user's company.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description ID of the Contact associated with this account. The contact must have a value in the <code>AccountId</code> field or an error occurs.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country associated with the user. Up to 80 characters allowed.</p>

Field	Details
CountryCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO country code associated with the user.</p>
CurrentStatus	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Text that describes what the user is working on.</p> <p> Note: If you update this field, the API automatically adds a post of type <code>UserStatus</code> on the user's profile in Chatter. This field is deprecated in API version 25.0. To achieve similar behavior, post to the user directly by creating a <code>FeedItem</code> with the user's <code>ParentId</code>.</p>
DefaultCurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Nillable, Restricted picklist, Update</p> <p>Description The user's default currency setting for new records. For example, a user in France could have a <code>DefaultCurrencyIsoCode</code> set to Euros, and that would be their default currency in the application. Only applicable for organizations that use multiple currencies.</p>
DefaultDivision	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description This record's default division. Only applicable if divisions are enabled for your organization.</p>
DefaultGroupNotificationFrequency	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Required. The default frequency for sending the user's Chatter group email notifications when the user joins groups. The valid values are:</p> <ul style="list-style-type: none">• P—Email on every post• D—Daily digests• W—Weekly digests• N—Never <p>The default value is N. For Professional, Enterprise, Unlimited, and Developer Edition organizations that existed prior to API version 22.0, the default value remains D.</p> <p>This field is available in API version 21.0 and later.</p>
DelegatedApproverId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Id of the user who is a delegated approver for this user.</p>
Department	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The company department associated with the user.</p>
DigestFrequency	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Required. The frequency at which the system sends the user's Chatter personal email digest. The valid values are:</p> <ul style="list-style-type: none">• D = Daily• W = Weekly• N = Never <p>The default value is D.</p>

Field	Details
Division	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The division associated with this user, similar to Department and unrelated to DefaultDivision.</p>
Email	<p>Type email</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Required. The user's email address.</p>
EmailEncodingKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. The email encoding for the user, such as ISO-8859-1 or UTF-8.</p>
EmailPreferencesAutoBcc	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether the user receives copies of sent emails. This option applies only if compliance BCC emails are not enabled.</p>
EmailPreferencesAutoBccStayInTouch	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Determines whether the user receives copies of sent Stay-in-Touch emails. This option applies only if compliance BCC emails are not enabled.</p>
EmailPreferencesStayInTouchReminder	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Filter, Update</p> <p>Description Determines whether the user is reminded to use the Stay-in-Touch feature when creating a new contact.</p>
EmployeeNumber	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's employee number.</p>
Extension	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's phone extension number.</p>
Fax	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's fax number.</p>
FederationIdentifier	<p>Type string</p> <p>Properties Create, Filter, idLookup, Nillable, Sort, Update</p> <p>Description Indicates the value that must be listed in the <code>Subject</code> element of a Security Assertion Markup Language (SAML) <i>IDP certificate</i> to authenticate the user for a client application using single sign-on. This value must be specified if the <code>SAML User ID Type</code> is Assertion contains Federation ID from the User record. Otherwise, this field can't be edited.</p>
FirstName	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's first name.</p>
ForecastEnabled	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the user is enabled as a Forecast Manager (<code>true</code>) or not (<code>false</code>) in customizable forecasting. Forecast managers see forecast rollups from users below them in the forecast hierarchy.</p>
FullPhotoUrl	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the user's profile photo. This field is available even if Chatter is disabled. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo. This field is available in API version 20.0 and later.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the user has access to log in (<code>true</code>) or not (<code>false</code>). You can modify a User's active status from the user interface or via the API.</p>
IsPartner	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p>

Field	Details
	<p>Description Indicates whether the user is a partner who has access to the partner portal (<code>true</code>) or not (<code>false</code>). This field is not available for release 9.0 and later. Instead, use <code>UserType</code> with the value <code>Partner</code> or <code>Power Partner</code>.</p>
IsPortalEnabled	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the user has access to Communities or portals (<code>true</code>) or not (<code>false</code>). This field is only available if one of the following conditions is true:</p> <ul style="list-style-type: none">• Communities are enabled in your organization and you have community or portal user licenses• Portals are enabled in your organization
IsPortalSelfRegistered	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the user is a Customer Portal user who self-registered for your organization's Customer Portal (<code>true</code>) or not (<code>false</code>). This field is not available for release 9.0 and earlier.</p>
IsPrmSuperUser	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Available for partner portal users only. Indicates whether the user has super user access in the partner portal (<code>true</code>) or not (<code>false</code>). This field is available in API version 24.0 and later.</p> <p> Note: This field is not automatically enabled. Contact Salesforce to enable this field.</p>
IsProfilePhotoActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description Indicates whether a user has a profile photo (<code>true</code>) or not (<code>false</code>). This field is available in API version 36.0 and later.</p>
JigsawImportLimitOverride	<p>Type <code>int</code></p> <p>Properties Create, Filter, Group Nillable, Sort, Update</p> <p>Description The Data.com user's monthly addition limit. The value must be between zero and the organization's monthly addition limit. Label is Data.com Monthly Addition Limit. This field is available in API version 27.0 and later.</p>
LanguageLocaleKey	<p>Type <code>picklist</code></p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. The user's language, such as "French" or "Chinese (Traditional)." Label is Language.</p>
LastLoginDate	<p>Type <code>dateTime</code></p> <p>Properties Filter, Sort, Nillable</p> <p>Description The date and time when the user last successfully logged in. This value is updated if 60 seconds have elapsed since the user's last login.</p>
LastName	<p>Type <code>string</code></p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. The user's last name.</p>
LastReferencedDate	<p>Type <code>date</code></p> <p>Properties Filter, Nillable, Sort, Update</p>

Field	Details
	<p>Description The timestamp for when the current user last viewed a record related to this record.</p>
LastViewedDate	<p>Type date</p> <p>Properties Filter, Nullable, Sort, Update</p> <p>Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.</p>
Latitude	<p>Type double</p> <p>Properties Create, Filter, Nullable, Sort, Update</p> <p>Description Used with Longitude to specify the precise geolocation of an address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
LocaleSidKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. This field is a restricted picklist field. The value of the field affects formatting and parsing of values, especially numeric values, in the user interface. It does not affect the API. The field values are named according to the language, and country if necessary, using two-letter ISO codes. The set of names is based on the ISO standard. It can often be more convenient to manually set a user's locale in the user interface, and then use that value for inserting or updating other users via the API.</p>
Longitude	<p>Type double</p> <p>Properties Create, Filter, Nullable, Sort, Update</p>

Field	Details
	<p>Description Used with <code>Latitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
Manager	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description User lookup field used to select the user's manager. This establishes a hierarchical relationship, preventing you from selecting a user that directly or indirectly reports to itself.</p>
ManagerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The Id of the user who manages this user.</p>
MediumBannerPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the medium sized user profile banner photo.</p>
MiddleName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's middle name. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
MobilePhone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description The user's mobile or cellular phone number.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Concatenation of FirstName and LastName. Limited to 121 characters.</p>
OfflineTrialExpirationDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date and time when the user's Connect Offline trial expires.</p>
Phone	<p>Type phone</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's phone number.</p>
PortalRole	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The role of the user in the Customer Portal (either Executive, Manager, User, or PersonAcount). Prior to API version 16.0, you could set this field to null and the system automatically included a portal role. In API version 16.0 and above, when you set this field to null, a portal role is not automatically created. When this field is null and a ContactId is provided, the user is assigned to the User role. The field is available if Customer Portal is enabled OR Communities is enabled and have available partner portal, Customer Portal, or High-Volume Portal User licenses.</p>

Field	Details
PostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's postal or ZIP code. Label is Zip/Postal Code.</p>
ProfileId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. ID of the user's Profile. Use this value to cache metadata based on profile. In earlier releases, this was <code>RoleId</code>.</p>
ReceivesAdminInfoEmails	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the user receives email for administrators from Salesforce (<code>true</code>) or not (<code>false</code>).</p>
ReceivesInfoEmails	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the user receives informational email from Salesforce (<code>true</code>) or not (<code>false</code>).</p>
SenderEmail	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The email address used as the From address when the user sends emails. This is the same value shown in Setup on the My Email Settings page.</p>
SenderName	<p>Type string</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The name used as the email sender when the user sends emails. This is the same value shown in Setup on the My Email Settings page.</p>
Signature	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The signature text added to emails. This is the same value shown in Setup on the My Email Settings page.</p>
SmallBannerPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the small user profile banner photo.</p>
SmallPhotoUrl	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for a thumbnail of the user's profile photo. This field is available even if Chatter is disabled. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo. This field is available in API version 20.0 and later.</p>
State	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state associated with the User. Up to 80 characters allowed.</p>

Field	Details
StateCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ISO state code associated with the user.</p>
StayInTouchNote	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The body specified for the Stay-in-Touch email.</p>
StayInTouchSignature	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The signature specified for the Stay-in-Touch email.</p>
StayInTouchSubject	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The subject specified for the Stay-in-Touch email.</p>
Street	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street address associated with the User.</p>
Suffix	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p>Description The user's name suffix. Maximum size is 40 characters. Contact Salesforce Customer Support to enable this field.</p>
TimeZoneSidKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. This field is a restricted picklist field. A User time zone affects the offset used when displaying or entering times in the user interface. However, the API does not use a User time zone when querying or setting values. Values for this field are named using region and key city, according to ISO standards. It can often be more convenient to manually set one User time zone in the user interface, and then use that value for creating or updating other User records via the API.</p>
Title	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user's business title, such as "Vice President."</p>
Username	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Required. Contains the name that a user enters to log into the API or the user interface. The value for this field must be in the form of an email address. It must also be unique across all organizations. If you try to create or update a User with a duplicate value for this field, the operation is rejected. Each inserted User also counts as a license. Every organization has a maximum number of licenses. If you attempt to exceed the maximum number of licenses by inserting User records, the create is rejected.</p>
UserPermissionsCallCenterAutoLogin	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
UserPermissionsChatterAnswersUser	Description Required if Salesforce CRM Call Center is enabled. Indicates whether the user is enabled to use the auto login feature of the call center (<code>true</code>) or not (<code>false</code>).
UserPermissionsInteractionUser	Type boolean
UserPermissionsJigsawProspectingUser	Properties Create, Filter, Update
UserPermissionsKnowledgeUser	Description Indicates whether the portal user is enabled to use the Chatter Answers feature (<code>true</code>) or not (<code>false</code>). This field defaults to <code>false</code> when a Customer Portal user is created from the API.
UserPermissionsLiveAgentUser	Type boolean
UserPermissionsKnowledgeUser	Properties Create, Filter, Update
UserPermissionsLiveAgentUser	Description Indicates whether the user is allocated one Data.com user license (<code>true</code>) or not (<code>false</code>). The Data.com user license lets the user add Data.com contact and lead records to Salesforce in supported editions. Label is Data.com User .
UserPermissionsKnowledgeUser	Type boolean

Field	Details
	Properties
	Create, Filter, Update
	Description
	Indicates whether the user is enabled to use Live Agent (<code>true</code>) or not (<code>false</code>). Label is Live Agent User .
UserPermissionsMarketingUser	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	Required. Indicates whether the user is enabled to manage campaigns in the user interface (<code>true</code>) or not (<code>false</code>). Label is Marketing User .
UserPermissionsMobileUser	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	Indicates whether the user is allocated one Salesforce Classic Mobile license (<code>true</code>) or not (<code>false</code>). Label is Apex Mobile User . The Salesforce Classic Mobile license grants the user access to the Salesforce Classic Mobile application on supported mobile devices.
UserPermissionsOfflineUser	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	Required. Indicates whether the user is enabled to use Offline Edition (<code>true</code>) or not (<code>false</code>). Label is Offline User .
UserPermissionsSFContentUser	Type
	boolean
	Properties
	Create, Filter, Update
	Description
	Indicates whether the user is allocated one Salesforce CRM Content User License (<code>true</code>) or not (<code>false</code>). Label is Salesforce CRM Content User . The Salesforce CRM Content User license grants the user access to the Salesforce CRM Content application.

Field	Details
UserPermissionsSiteforceContributorUser	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether the user is allocated one Site.com Contributor feature license (<code>true</code>) or not (<code>false</code>). Label is Site.com Contributor User. The Site.com Contributor feature license grants the user access to the Site.com application. Users with a Contributor license can use Site.com Studio to edit site content only.</p>
UserPermissionsSiteforcePublisherUser	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether the user is allocated one Site.com Publisher feature license (<code>true</code>) or not (<code>false</code>). Label is Site.com Publisher User. The Site.com Publisher feature license grants the user access to the Site.com application. Users with a Publisher license can build and style websites, control the layout and functionality of pages and page elements, and add and edit content.</p>
UserPermissionsSupportUser	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the user can use the Salesforce console.</p>
UserPermissionsWirelessUser	<p>Type boolean</p> <p>Properties Create, Update</p> <p>Description Required if the Wireless permission is enabled for your organization. Indicates whether the user is enabled to use Wireless Edition (<code>true</code>) or not (<code>false</code>). Label is Wireless User.</p>
	 Note: As of November 2005, Salesforce Wireless Edition is no longer available for purchase. If you are a Professional Edition customer and purchased Wireless Edition prior to November 7, 2005 or are an Enterprise Edition customer who has signed or renewed your Salesforce contract prior to November 7, 2005,

Field	Details
	<p>you may continue using Wireless Edition through the end of your existing contract term.</p>
UserPermissionsWorkDotComUserFeature	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether the Work.com feature is enabled for the user (<code>true</code>) or not (<code>false</code>).</p>
UserPreferencesActivityRemindersPopup	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, a reminder popup window automatically opens when an activity reminder is due. Corresponds to the <code>Trigger alert when reminder comes due</code> checkbox at the Reminders page in the personal settings in the user interface.</p>
UserPreferencesApexPagesDeveloperMode	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, indicates the user has enabled developer mode for editing Visualforce pages and controllers.</p>
UserPreferencesContentEmailAsAndWhen	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, a user with Salesforce CRM Content subscriptions receives a once daily email summary if activity occurs on his or her subscribed content, libraries, tags, or authors. To receive email, the <code>UserPreferencesContentNoEmail</code> field must also be <code>false</code>. The default value is <code>false</code>.</p> <p> Note: This field is only visible when Salesforce CRM Content is enabled for your organization.</p>

Field	Details
UserPreferencesContentNoEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, a user with Salesforce CRM Content subscriptions receives email notifications if activity occurs on his or her subscribed content, libraries, tags, or authors. To receive real-time email alerts, set this field to <code>false</code> and set the <code>UserPreferencesContentEmailAsAndWhen</code> field to <code>true</code>. The default value is <code>false</code>.</p> <p> Note: This field is only visible when Salesforce CRM Content is enabled for your organization.</p>
UserPreferencesEnableAutoSubForFeeds	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the user automatically subscribes to feeds for any objects that the user creates. This field is available in API version 25.0 and later.</p>
UserPreferencesDisableAllFeedsEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email for all updates to Chatter feeds, based on the types of feed emails and digests the user has enabled. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableAutoSubForFeeds	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically subscribes to feeds for any objects that the user creates. This field is deprecated in API version 25.0 and later. Starting with API version 25.0, use <code>UserPreferencesEnableAutoSubForFeeds</code> to enable or disable auto-follow for objects a user creates.</p>

Field	Details
UserPreferencesDisableBookmarkEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a Chatter feed item after the user has bookmarked it. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableChangeCommentEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a change the user has made, such as an update to their profile. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableEndorsementEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the member automatically receives email every time someone endorses them for a topic.</p>
UserPreferencesDisableFileShareNotificationsForApi	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, email notifications are sent from the person who has shared a file to the users with whom the file has been shared. This field is available in API version 25.0 and later.</p>
UserPreferencesDisableFollowersEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
UserPreferencesDisableLaterCommentEmail	<p>Description When <code>false</code>, the user automatically receives email every time someone starts following the user in Chatter. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableLikeEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a feed item after the user has commented on the feed item. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableMentionsPostEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time someone likes a post or comment the user has made. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableProfilePostEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time the user is mentioned in posts. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableSharePostEmail	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time the user's post is shared. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableFeedbackEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives emails related to Work.com feedback. This includes when someone requests or offers feedback, shares feedback with the user, or reminds the user to answer a feedback request.</p>
UserPreferencesDisCommentAfterLikeEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a post the user has liked. This field is available in API version 24.0 and later.</p>
UserPreferencesDisMentionsCommentEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email every time the user is mentioned in comments. This field is available in API version 24.0 and later.</p>
UserPreferencesDisableMessageEmail	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>false</code>, the user automatically receives email for Chatter messages sent to the user. This field is available in API version 24.0 and later.</p>

Field	Details
UserPreferencesDisableRewardEmail	Type boolean
	Properties Filter
	Description <p>When <code>false</code>, the user automatically receives emails related to Work.com rewards. This includes when someone gives a reward to the user.</p>
UserPreferencesDisableWorkEmail	Type boolean
	Properties Filter
	Description <p>When <code>false</code>, the user receives emails related to Work.com feedback, goals, and coaching. The user must also sign up for individual emails listed on the Work.com email settings page. When <code>true</code>, the user doesn't receive any emails related to Work.com feedback, goals, or coaching even if they are signed up for individual emails.</p>
UserPreferencesDisProfPostCommentEmail	Type boolean
	Properties Create, Filter, Update
	Description <p>When <code>false</code>, the user automatically receives email every time someone comments on posts on the user's profile. This field is available in API version 24.0 and later.</p>
UserPreferencesEventRemindersCheckboxDefault	Type boolean
	Properties Create, Filter, Update
	Description <p>When <code>true</code>, a reminder popup is automatically set on the user's events. Corresponds to the <code>By default, set reminder on Events to...</code> checkbox on the Reminders page in the user interface. This field is related to UserPreference and customizing activity reminders.</p>
UserPreferencesHideBiggerPhotoCallout	Type boolean

Field	Details
	<p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, users can choose to hide the callout text below the large profile photo.</p>
UserPreferencesHideChatterOnboardingSplash	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the initial Chatter onboarding prompts do not appear.</p>
UserPreferencesHideCSNDesktopTask	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the Chatter recommendations panel never displays the recommendation to install Chatter Desktop. This field is available in API version 26.0 and later.</p>
UserPreferencesHideCSNGetChatterMobileTask	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, the Chatter recommendations panel never displays the recommendation to install Chatter Mobile. This field is available in API version 26.0 and later.</p>
UserPreferencesHideEndUserOnboardingAssistantModal	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Reserved for future use.</p>
UserPreferencesHideLightningMigrationModal	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
UserPreferencesHideSecondChatterOnboardingSplash	<p>Description Reserved for future use.</p>
UserPreferencesHideS1BrowserUI	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the secondary Chatter onboarding prompts do not appear.</p>
UserPreferencesHideSfxWelcomeMat	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Controls the interface that the user sees when logging in to Salesforce from a supported mobile browser. If <code>false</code>, the user is automatically redirected to the Salesforce1 mobile browser app. If <code>true</code>, the user sees the full Salesforce site. The default value is <code>false</code>. Label is Salesforce1 User. This field is available in API version 29.0 or later.</p>
UserPreferencesJigsawListUser	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Controls whether a user sees the Lightning Experience new user message. That message welcomes users to the new interface and provides step-by-step instructions that describe how to return to Salesforce Classic.</p>

Field	Details
UserPreferencesLightningExperiencePreferred	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, redirects the user to the Lightning Experience interface. Label is Switch to Lightning Experience. This field is available in API version 35.0 and later.</p>
UserPreferencesOptOutOfTouch	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description This field is deprecated in API version 29.0. When <code>false</code>, the user automatically accesses the Salesforce Touch app when logging in to Salesforce from an iPad. If <code>true</code>, automatic access to the Salesforce Touch app is turned off and the user's iPad is directed to the full Salesforce site instead. The default value is <code>false</code>.</p> <p> Note: Salesforce Touch must be enabled before this field is visible.</p>
UserPreferencesPathAssistantCollapsed	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, Sales Path appears collapsed or hidden to the user. This field is available in API version 35.0 and later.</p>
UserPreferencesProcessAssistantCollapsed	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, Sales Path appears collapsed or hidden to the user. This field is available in API versions 33.0 and 34.0 only. In API versions 35.0 and later, use <code>UserPreferencesPathAssistantCollapsed</code>.</p>
UserPreferencesReminderSoundOff	<p>Type boolean</p>

Field	Details
	<p>Properties Create, Filter, Update</p> <p>Description When <code>true</code>, a sound automatically plays when an activity reminder is due. Corresponds to the <code>Play a reminder</code> sound checkbox on the Reminders page in the user interface.</p>
<code>UserPreferencesShowCityToExternalUsers</code>	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the city field in the user's contact information. City is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none">• This field is <code>false</code>. When <code>false</code>, this field returns the value <code>#N/A</code>. <p>City is visible to external members in a community when:</p> <ul style="list-style-type: none">• This field is <code>true</code>, or• This field is <code>false</code> but <code>UserPreferencesShowCityToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
<code>UserPreferencesShowCityToGuestUsers</code>	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the city field in the user's contact information. When <code>true</code>, city is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value <code>#N/A</code>.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowCityToExternalUsers</code>, making the user's city visible to external members.</p> <p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>

Field	Details
UserPreferencesShowCountryToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the country field in the user's contact information. Country is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none"> This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A. <p>Country is visible to external members in a community when:</p> <ul style="list-style-type: none"> This field is <code>true</code>, or This field is <code>false</code> but <code>UserPreferencesShowCountryToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowCountryToGuestUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the country field in the user's contact information. When <code>true</code>, country is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowCountryToExternalUsers</code>, making the user's country visible to external members.</p> <p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
UserPreferencesShowEmailToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
	<p>Description</p> <p>Indicates the visibility of the email address field in the user's contact information. Email address is visible only to internal members of the user's organization when this field is <code>false</code>. Email address is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowEmailToGuestUsers	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the email address field in the user's contact information. When <code>true</code>, the email address is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowEmailToExternalUsers</code>, making the user's email address visible to guests.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>
UserPreferencesShowFaxToExternalUsers	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the fax number field in the user's contact information. Fax number is visible only to internal members of the user's organization when this field is <code>false</code>. Fax number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowFaxToGuestUsers	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Filter, Update</p>

Field	Details
	<p>Description</p> <p>Indicates the visibility of the fax number field in the user's contact information. When <code>true</code>, the fax number field is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowFaxToExternalUsers</code>, making the user's fax number visible to guests.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>
<code>UserPreferencesShowManagerToExternalUsers</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the manager field in the user's contact information. Manager is visible only to internal members of the user's organization when this field is <code>false</code>. Manager is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
<code>UserPreferencesShowManagerToGuestUsers</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the manager field in the user's contact information. When <code>true</code>, the manager field is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowManagerToExternalUsers</code>, making the user's manager visible to guests.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>
<code>UserPreferencesShowMobilePhoneToExternalUsers</code>	<p>Type</p> <p>boolean</p>

Field	Details
	<p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the mobile or cellular phone number field in the user's contact information. The number is visible only to internal members of the user's organization when this field is <code>false</code>. The number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowMobilePhoneToGuestUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the mobile phone field in the user's contact information. When <code>true</code>, the mobile phone field is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowMobilePhoneToExternalUsers</code>, making the user's mobile phone visible to guests. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>
UserPreferencesShowPostalCodeToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the postal or ZIP code field in the user's contact information. Postal code is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none">• This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A. <p>Postal code is visible to external members in a community when:</p> <ul style="list-style-type: none">• This field is <code>true</code>, or

Field	Details
UserPreferencesShowPostalCodeToGuestUsers	<ul style="list-style-type: none">This field is <code>false</code> but <code>UserPreferencesShowPostalCodeToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowProfilePicToGuestUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the postal or ZIP code field in the user's contact information. When <code>true</code>, postal code is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A. When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowPostalCodeToExternalUsers</code>, making the user's postal code visible to external members. The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
UserPreferencesShowStateToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the user's profile photo. When <code>true</code>, the photo is visible to guest users in a community. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the stock photo. The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>

Field	Details
UserPreferencesShowStateToGuestUsers	<ul style="list-style-type: none">This field is <code>false</code>. When <code>false</code>, this field returns the value <code>#N/A</code>. <p>State is visible to external members in a community when:</p> <ul style="list-style-type: none">This field is <code>true</code>, orThis field is <code>false</code> but <code>UserPreferencesShowStateToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowStreetAddressToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the state field in the user's contact information. When <code>true</code>, state is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value <code>#N/A</code>. When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowStateToExternalUsers</code>, making the user's state visible to external members. The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the street address field in the user's contact information. The address is visible only to internal members of the user's organization when this field is <code>false</code>. The address is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>

Field	Details
UserPreferencesShowStreetAddressToGuestUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the street address field in the user's contact information. When <code>true</code>, the street address field is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in UserPreferencesShowStreetAddressToExternalUsers, making the user's street address visible to guests.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>
UserPreferencesShowTitleToExternalUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates the visibility of the business title field in the user's contact information. Title is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none"> This field is <code>false</code>. When <code>false</code>, this field returns the value <code>#N/A</code>. <p>Title is visible to external members in a community when:</p> <ul style="list-style-type: none"> This field is <code>true</code>, or This field is <code>false</code> but UserPreferencesShowTitleToGuestUsers is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>true</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowTitleToGuestUsers	<p>Type boolean</p> <p>Properties Create, Filter, Update</p>

Field	Details
	<p>Description</p> <p>Indicates the visibility of the business title field in the user's contact information. When <code>true</code>, title is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowTitleToExternalUsers</code>, making the user's title visible to external members.</p> <p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
<code>UserPreferencesShowWorkPhoneToExternalUsers</code>	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the work phone number field in the user's contact information. The number is visible only to internal members of the user's organization when this field is <code>false</code>. The number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
<code>UserPreferencesShowWorkPhoneToGuestUsers</code>	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description</p> <p>Indicates the visibility of the work phone field in the user's contact information. When <code>true</code>, the work phone field is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowWorkPhoneToExternalUsers</code>, making the user's work phone visible to guests.</p> <p>When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 34.0 and later.</p>

Field	Details
UserPreferencesSortFeedByComment	Type boolean
	Properties Create, Filter, Update
	Description
	<p>Specifies the data value used in sorting a user's feed. When <code>true</code>, the feed is sorted by most recent comment activity. When <code>false</code>, the feed is sorted by post date.</p>
UserPreferencesTaskRemindersCheckboxDefault	Type boolean
	Properties Create, Filter, Update
	Description
	<p>When <code>true</code>, a reminder popup is automatically set on the user's tasks. Corresponds to the <code>By default, set reminder on Tasks to...</code> checkbox on the Reminders page in the user interface. This field is related to UserPreference and customizing activity reminders.</p>
UserRoleID	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description
	ID of the user's UserRole. Label is Role ID .
UserType	Type picklist
	Properties Filter, Group, Nillable, Sort, Restricted picklist
	Description
	<p>The category of user license. Each <code>UserType</code> is associated with one or more <code>UserLicense</code> records. Each <code>UserLicense</code> is associated with one or more profiles. In API version 10.0 and later, valid values include:</p>
	<ul style="list-style-type: none"> • Standard: user license. This user type also includes Salesforce Platform and Salesforce Platform One user licenses. Label is Standard. • PowerPartner: User whose access is limited because he or she is a partner and typically accesses the application through a partner portal or community. Label is Partner.

Field	Details
WirelessEmail	<ul style="list-style-type: none"> CSPLitePortal: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal or community. Label is High Volume Portal. CustomerSuccess: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal User. PowerCustomerSuccess: user whose access is limited because he or she is an organization's customer and accesses the application through a Customer Portal. Label is Customer Portal Manager. Users with this license type can view and edit data they directly own or data owned by or shared with users below them in the Customer Portal role hierarchy. CsnOnly: user whose access to the application is limited to Chatter. This user type includes Chatter Free and Chatter moderator users. Label is Chatter Free.
	<p>Type email</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Wireless email address associated with this user. For use with Salesforce Wireless Edition. This field is available only if the Wireless and Email permissions are enabled for your organization.</p> <p> Note: As of November 2005, Salesforce Wireless Edition is no longer available for purchase. If you are a Professional Edition customer and purchased Wireless Edition prior to November 7, 2005 or are an Enterprise Edition customer who has signed or renewed your Salesforce contract prior to November 7, 2005, you may continue using Wireless Edition through the end of your existing contract term.</p>

Usage

Use this object to query information about users and to provision and modify users in your organization. Unlike other objects, the records in the User table represent actual users—not data owned by users. Any user can query or describe User records.

For example, the following SOQL code finds users with a particular user role.

```
SELECT Id, Username
FROM User
WHERE UserRoleID='00ED000000xicT'
```

Each portal user is associated with a portal account. A portal account can have a maximum of three portal roles (Executive, Manager, and User). You can select the default number of roles to be created from the user interface. The role hierarchy is maintained when you

insert and delete portal roles, and roles are created bottom-up. Deleting the User role causes the Manager role to be renamed to User role. Deleting both the Executive and User roles causes the Manager role to be renamed to User role. Before deleting a role, you must assign users under that role to another role.

Deactivate Users

You can't delete a user in the user interface or the API. You can deactivate a user in the user interface; and you can deactivate or disable a Customer Portal or partner portal user in the user interface or the API. Because users can never be deleted, we recommend that you exercise caution when creating them.

If you deactivate a user, any EntitySubscription where the user is associated with the ParentId or SubscriberId field, meaning all subscriptions both to and from the user, are soft deleted. If the user is reactivated, the subscriptions are restored. However, if you deactivate multiple users at once and these users follow each other, their subscriptions are hard deleted. In this case, the user-to-user EntitySubscription is deleted twice (double deleted). Such subscriptions can't be restored upon user reactivation.

Passwords

For security reasons, you can't query User passwords via the API or the user interface. However, the API allows you to set and "reset" User passwords using the `setPassword()` and `resetPassword()` calls. The password lockout status and the ability to reset the User locked-out status is not available via the API. You must check and reset the User password lockout status using the user interface.

SEE ALSO:

[getUserInfo\(\)](#)
[create\(\)](#)
[update\(\)](#)
[query\(\)](#)
[search\(\)](#)
[retrieve\(\)](#)
[upsert\(\)](#)
[update\(\)](#)
[getUpdated\(\)](#)
[getDeleted\(\)](#)
[describeSObjects\(\)](#)
[Frequently-Occurring Fields](#)
[UserRole](#)
[UserLicense](#)

UserAccountTeamMember

Represents a User on the default account team of another User.

See also OpportunityTeamMember, which represents a User on the opportunity team of an Opportunity

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
AccountAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description Required. For Account records that the user has added to his or her default account team, the level of access the account team member has. . The possible values are:</p> <ul style="list-style-type: none">• Read• Edit <p>This field must be set to an access level that is higher than the organization's default access level for accounts.</p>
CaseAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description Required. Level of access that the account team member has to Case records related to the account. The possible values are:</p> <ul style="list-style-type: none">• None• Read• Edit <p>This field must be set to an access level that is higher than the organization's default access level for cases.</p>
ContactAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p>

Field	Details
	<p>Description Required. For Contact records related to the account, the level of access that the account team member has. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is higher than the organization's default access level for contacts.</p> <p> Note: When DefaultContactAccess is set to Controlled by Parent, you can't create or update this field.</p>
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p> <p>Description Required. Level of access that the team member has to Opportunity records related to the account. The possible values are:</p> <ul style="list-style-type: none"> • None • Read • Edit <p>This field must be set to an access level that is higher than the organization's default access level for opportunities.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the User who owns the default account team.</p>
TeamMemberRole	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description Role that the team member has on opportunities for which the user has added his or her default account team. The valid values are set by the organization's administrator in the Account Team Roles picklist. Label is Team Role.</p>

Field	Details
UserId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description Required. ID of the User who is a member of the default account team. This field cannot be updated.</p>

Usage

This object is available only in organizations that have enabled the account teams functionality, which can be done using the user interface.

If you attempt to create a record that matches an existing record, the create call updates any modified fields and returns the existing record.

You can set up a User record so the default account team includes the others who typically work with them on accounts.

UserApplInfo

Stores the last Lightning app logged in to. This object is available in API version 38.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
FormFactor	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The relative size of the app as displayed. Values are:</p> <ul style="list-style-type: none"> • Small—suitable for a small device like a mobile phone • Medium—suitable for a tablet • Large—suitable for a large display device, like a monitor <p>It's possible to have three versions of the app as the one last logged in to, where each version has a different form factor.</p>

Field Name	Details
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user that used this app.</p>

UserAppMenuCustomization

Represents an individual user's settings for items in the Force.com app menu or App Launcher. This object is available in API version 35.0 and later.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
ApplicationId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The 15-character ID for the application associated with the menu item.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The ID of the user for these specific settings.</p>
SortOrder	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	Description
	<p>The index value that controls where this item appears in the menu. For example, a menu item with a sort order value of 5 will appear between items with sort order values of 3 and 9.</p>

Usage

See the AppMenuItem object for the organization-wide default settings This object contains the fields representing any changes the user made to the menu.

UserAppMenuItemCustomizationShare

Represents a sharing entry on a UserAppMenuItemCustomization record. This object is available in API version 35.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the parent record, if any.</p>
RowCause	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Values may include:</p> <ul style="list-style-type: none"> • Manual—The User or Group has access because a user with “All” access manually shared the UserAppMenuCustomization record with them. • Owner—The User is the owner of the UserAppMenuCustomization record or is in a role above the UserAppMenuCustomization record owner in the role hierarchy.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the UserAppMenuCustomization record.</p>

UserAppMenuItem

Represents the organization-wide settings for items in the Force.com app menu or App Launcher that the requesting user has access to in Setup. This object is available in API version 35.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `query()`, `search()`

Fields

Field Name	Details
AppMenuItemId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 15-character ID for the menu item.</p>

Field Name	Details
ApplicationId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The 15-character ID for the application associated with the menu item.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A description of this menu item.</p>
IconUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The icon for the menu item's application.</p>
InfoUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL for more information about the application.</p>
IsUsingAdminAuthorization	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If true, the app is pre-authorized for certain users by the administrator.</p>
IsVisible	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>If true, the app is visible to the user.</p>
Label	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable</p> <p>Description</p> <p>The app's name.</p>
LogoUrl	<p>Type</p> <p>url</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The logo for the menu item's application. The default is the initials of the Label value.</p>
MobileStartUrl	<p>Type</p> <p>url</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The location mobile users are directed to after they've authenticated. This is only used with connected apps.</p>
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The API name of the item.</p>
SortOrder	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p>

Field Name	Details
Description	
	The index value that controls where this item appears in the menu. For example, a menu item with a sort order value of 5 will appear between items with sort order values of 3 and 9.
StartUrl	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The location users are directed to after they've authenticated. For a connected app, this is the location specified by the StartUrl. Otherwise it's the application's default start page.</p>
Type	
	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of application represented by this item. The types are:</p> <ul style="list-style-type: none"> • ConnectedApplication • ServiceProvider • TabSet
UserSortOrder	
	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The index value that represents where the user set this item in the menu (or App Launcher). For example, an item with a sort order value of 5 will appear between items with sort order values of 3 and 9. This value is separate from SortOrder so you can create logic incorporating both values. For example, if you want the user-sorted items to appear first, followed by the organization order for the rest, use:</p> <pre>SELECT ApplicationId, SortOrder, UserSortOrder FROM AppMenuItem order by userSortOrder NULLS LAST, sortOrder NULLS LAST</pre>

Usage

See the AppMenuItem object for the organization-wide default settings. This object contains the fields the requesting user has permission to see.

UserConfigTransferButton

Represents the association between a Live Agent configuration and a live chat button. This association allows users associated with a specific configuration to transfer chats to a button queue.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
LiveChatButtonId	Type reference Properties Create, Filter, Group, Nullable, Sort Description The ID of the live chat button that agents can transfer chats to.
LiveChatUserConfigId	Type reference Properties Create, Filter, Group, Sort Description The ID of the Live Agent configuration; agents associated with this configuration can transfer chats to the chat button indicated by the <code>LiveChatButtonId</code> .

UserConfigTransferSkill

Represents the association between a Live Agent configuration and a skill. This association allows users associated with a specific configuration to transfer chats to agents who have that skill.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

Field Name	Details
LiveChatUserConfigId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the Live Agent configuration; agents associated with this configuration can transfer chats to the chat button indicated by the <code>LiveChatButtonId</code>.</p>
SkillId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the skill group that agents can transfer chats to.</p>

UserCustomBadge

Represents a custom badge for a user. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
BadgeType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of badge. Valid values are:</p> <ul style="list-style-type: none"> • Customer • Partner • Employee

Field Name	Details
CustomText	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Custom text for the badge.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the community or org that the badge is in.</p>

UserCustomBadgeLocalization

Represents the translated version of a custom badge for a user. This object is available in API version 38.0 and later.

Supported Calls

`create(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update()`

Special Access Rules

- Translation Workbench must be enabled for your org.
- Users with the “Customize Application” or “Manage Translation” permission can create or update UserCustomBadge translations.

Fields

Field Name	Details
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The language the UserCustomBadge is translated into. This picklist contains these fully supported languages.</p> <ul style="list-style-type: none"> Chinese (Simplified): zh_CN Chinese (Traditional): zh_TW

Field Name	Details
	<ul style="list-style-type: none">• Danish: da• Dutch: nl_NL• English: en_US• Finnish: fi• French: fr• German: de• Italian: it• Japanese: ja• Korean: ko• Norwegian: no• Portuguese (Brazil): pt_BR• Russian: ru• Spanish: es• Spanish (Mexico): es_MX• Swedish: sv• Thai: th
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none">• In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.• In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description ID of the UserCustomBadge.</p>
Value	<p>Type string</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The translated text for the UserCustomBadge. Label is Translation Text.</p>

UserFeed

Represents a single feed item in the feed displayed on a Chatter user profile feed. A user profile feed shows changes to a user record for fields that are tracked in feeds, and posts and comments about the record. This object is available in API version 18.0 and later.

A user feed shows recent changes to a user record for any fields that are tracked in feeds, and posts and comments about the record. It is a useful way to stay up-to-date with changes made to users in the organization.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

If you use the FeedComment object to comment on a user record, the user can delete the comment. For example, if John Smith makes a comment about Sasha Jones, Sasha can delete the comment.

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the User object
- "Moderate Chatter"



Note: Users with the "Moderate Chatter" permission can delete only the feed items and comments they see.

Fields

Field	Details
Body	<p>Type textarea</p>

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The content of UserFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission.</p> <p>This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description When a PartnerNetworkConnection modifies a record that is tracked, the CreatedBy field contains the ID of the system administrator. The ConnectionId contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>

Field	Details
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by CreatedDateDESC sorts the feed by the most recent feed item.</p>

Field	Details
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code>

Field	Details
	<ul style="list-style-type: none"> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LastModifiedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type picklist</p>

Field	Details
	<p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none">• <code>NetworkId</code>—The ID of the community in which the <code>FeedItem</code> is available. If left empty, the feed item is only available in the default community.• <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>.• For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>.• You can't filter a <code>FeedItem</code> on the <code>NetworkScope</code> field.
<code>ParentId</code>	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
<code>RelatedRecordId</code>	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the <code>ContentVersion</code> object associated with a <code>ContentPost</code>. This field is null for all posts except <code>ContentPost</code>.</p>
<code>Title</code>	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the user record. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL and this field is the link name.</p>

Field	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • CaseCommentPost—generated event when a user adds a case comment for a case object

Field	Details
	<ul style="list-style-type: none">• <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received• <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• <code>ChangeStatusPost</code>—generated event when a user changes the status of a case• <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none">• <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item.• <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none">• For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default.• External users can set <code>Visibility</code> only to <code>AllUsers</code>.• On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to track changes for a user record.

SEE ALSO:

[User](#)
[EntitySubscription](#)
[NewsFeed](#)
[UserProfileFeed](#)

UserLicense

Represents a user license in your organization. A user license entitles a user to specific functionality and determines the profiles and permission sets available to the user.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field	Details
LicenseDefinitionKey	<p>Type string</p> <p>Properties Filter, Nillable</p> <p>Description A string that uniquely identifies a particular user license. Label is <code>License Def.</code> ID. Values are: AUL: corresponds to the Salesforce Platform user license AUL1: corresponds to the Salesforce Platform One user license AUL_LIGHT corresponds to the Salesforce Platform Light user license FDC_ONE corresponds to the Force.com - One App user license FDC_SUB corresponds to the Force.com App Subscription user license High Volume Customer Portal User: corresponds to the High Volume Customer Portal user license Overage_Platform_Portal_User corresponds to the Overage Authenticated Website user license PID_STRATEGIC_PRM: corresponds to the Gold Partner user license PID_CHATTER corresponds to the Chatter Only user license PID_CONTENT corresponds to the Content Only user license PID_Customer_Portal_Basic: corresponds to the Customer Portal Manager Standard user license and the Customer Portal User license PID_Customer_Portal_Standard: corresponds to the Customer Portal Manager Custom user license PID_FDC_FREE corresponds to the Force.com Free user license PID_IDEAS corresponds to the Ideas Only user license PID_Ideas_Only_Portal corresponds to the Ideas Only Portal user license PID_Ideas_Only_Site corresponds to the Ideas Only Site user license PID_KNOWLEDGE corresponds to the Knowledge Only user license</p>

Field	Details
	<p>PID_Customer_Community corresponds to the Customer Community license.</p> <p>PID_Customer_Community_Login corresponds to the Customer Community Login license.</p> <p>PID_Partner_Community corresponds to the Partner Community license.</p> <p>PID_Partner_Community_Login corresponds to the Partner Community Login license.</p> <p>PID_Limited_Customer_Portal_Basic: corresponds to the Limited Customer Portal Manager Standard user license</p> <p>PID_Limited_Customer_Portal_Standard: corresponds to the Limited Customer Portal Manager Custom user license</p> <p>PID_Overage_Customer_Portal_Basic: corresponds to the Overage Customer Portal Manager Standard user license</p> <p>PID_Overage_High_Volume_Customer_Portal corresponds to the Overage High Volume Customer Portal user license</p> <p>Platform_Portal_User: corresponds to the Authenticated Website user license</p> <p>POWER_PRM: corresponds to the Partner user license</p> <p>POWER_SSP: corresponds to the Customer Portal Manager user license</p> <p>SFDC: corresponds to the Full CRM user license</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The user license label. This field is available in API version 32.0 and later. This field is available through the API Access to User Licenses pilot program. For information on enabling this pilot program for your organization, contact Salesforce.</p>
MonthlyLoginsEntitlement	<p>Type int</p> <p>Properties Filter, Nillable</p> <p>Description The maximum number of customer or partner portal logins allowed per month. A <code>null</code> value in this field means the user license is charged according to the number of users rather than the number of logins. This field is available in API version 20.0 and later.</p>
MonthlyLoginsUsed	<p>Type int</p>

Field	Details
	<p>Properties Nillable</p> <p>Description The number of successful logins for all users associated with a customer or partner portal user license. This field has a non-null value if <code>MonthlyLoginsEntitlement</code> has a non-null value. This field is available in API version 20.0 and later.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The internal name of the user license.</p> <p> Note: Your organization may also include custom user licenses.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The current status of the user license. Valid values for this field are <code>Active</code> and <code>Disabled</code>. This field is available in API version 32.0 and later. This field is available through the API Access to User Licenses pilot program. For information on enabling this pilot program for your organization, contact Salesforce.</p>
TotalLicenses	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of user licenses in the organization. This field is available in API version 32.0 and later. This field is available through the API Access to User Licenses pilot program. For information on enabling this pilot program for your organization, contact Salesforce.</p>
UsedLicenses	<p>Type int</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description The number of user licenses that are assigned to active users in the organization.</p> <p>This field is available in API version 32.0 and later.</p> <p>This field is available through the API Access to User Licenses pilot program. For information on enabling this pilot program for your organization, contact Salesforce.</p>

Usage

Users with the "View Setup and Configuration" permission can use the UserLicense object to view the set of currently defined user licenses in your organization.

The UserLicense object is currently used by bulk user creation to determine the user license to which each profile and permission set belongs. For example, if you use the API to create portal users and you want to know which profile belongs to each portal user license, you can query this object for each profile and check the `LicenseDefinitionKey` to identify the associated user license.

SEE ALSO:

[Profile](#)

UserLogin

Represents the settings that affect a user's ability to log into an organization. To access this object, you need the `UserPermissions.ManageUsers` permission. This object is available in API version 29.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
<code>IsFrozen</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, the user account associated with this object is frozen.</p>
<code>IsPasswordLocked</code>	<p>Type boolean</p>

Field Name	Details
	<p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If <code>true</code>, the user account associated with this object is locked because of too many login failures. From the API, you can set this field to <code>false</code>, but not <code>true</code>.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the associated user account. This field can't be updated.</p>

Usage

To query for all frozen users in your organization:

```
SELECT Id, UserId
FROM UserLogin
WHERE IsFrozen = true
```

UserMembershipSharingRule

Represents the rules for sharing user records from a source group to a target group. A user record contains details about a user. Users who are members of the source group can be shared with members of the target group. The source and target groups can be based on roles, portal roles, public groups, or territories. This object is available in API version 26.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters. This field is available in API version 29.0 and later.</p>

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Rule Name in the user interface.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label on the user interface.</p>
UserAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing being allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
UserOrGroupId	<p>Type reference</p>

Field	Details
Properties	Create, Filter, Group, Sort
Description	The ID representing the target group being given access.

Usage

Use this object to manage sharing rules for user records. Source and target groups can include internal users, portal users, Chatter or Chatter External users.

UserPackageLicense

Represents a license for an installed managed package, assigned to a specific user. This object is available in API version 31.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field Name	Details
PackageLicenseId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The 18-character Globally Unique ID (GUID) that identifies the package license</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The User ID of the user licensed to use this package</p>

Usage

Use this object, in conjunction with PackageLicense, to provide users access to a managed package installed in your organization.

UserPreference

Represents a functional preference for a specific user in your organization.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
Preference	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the user preference. Supported values are Event Reminder Default Lead Time, Task Reminder Default Time, Prevent Logs on Load, Autocomplete Apex After Key Press, Visualforce Viewstate Inspector, Forecasting Displayed Type, Editor Theme, Editor Font Size, Pinned Folders, Enable Query Plan, and Email Transport Type. These values are related to UserPreferencesEventRemindersCheckboxDefault and UserPreferencesTaskRemindersCheckboxDefault on the User object. Enable New Open Dialog is reserved for future use.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the user associated with this role. Label is User ID.</p>
Value	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	Description
	The value of the user preference. For <code>Event Reminder Default Lead Time</code> , the values are increasing intervals of time from 0 minutes to 2 days. For <code>Task Reminder Default Time</code> , the values are half-hours from 12:00 AM to 11:30 PM. To view the respective sets of values, access the Reminders in your personal settings in the online application.

Usage

Use this object to query the set of currently configured user preferences in your organization. In your client application, you can query the User object to obtain valid User IDs in order to access the UserPreference object.

All users can invoke `query()` or `describeSObjects()` with this object.

UserProfile

Represents a Chatter user profile.



Note: This object has been deprecated as of API version 32.0. Use the [User](#) object to query information about a user in API version 32.0 and later.

Supported Calls

`describeLayout()`, `query()`, `retrieve()`

Special Access Rules

- Information in hidden fields in a user's profile is not searchable by external users (with a portal profile) in a community. For example, if a user in a community has a hidden email address and an external user searches for it, the user record is not returned in the search results. Hidden field values are also not returned when external users perform searches on non-hidden fields. So if an external user searches for a user's name (cannot be hidden), any hidden field values associated with the user record (for example, a hidden email address) are not returned in the search results.
- Internal users from your organization who belong to the same community, however, can both search for and view hidden field values in search results.
- Any fields that have been restricted in visibility will be returned empty, whether or not they are, and will not be removed from the field listing.

Fields

Field	Details
	Type
<code>AboutMe</code>	textarea

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Information about the user, such as areas of interest or skills.</p>
Address (beta)	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address. Read-only. See Address Compound Fields for details on compound address fields.</p>
City	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The city associated with the user profile.</p>
CompanyName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The company associated with the user profile.</p>
Country	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The country associated with the user profile.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The email address associated with the user profile.</p>

Field	Details
Fax	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The fax number associated with the user profile.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user's first name.</p>
FullPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for the user's profile photo if Chatter is enabled. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the user has access to log in (<code>true</code>) or not (<code>false</code>). You can modify a User's active status from the user interface or via the API.</p>
IsBadged	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description Indicates whether the user is visually badged (<code>true</code>) or not (<code>false</code>). Users of the same Chatter user type (internal, external) are badged. Different user types are not badged.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The user's last name.</p>
Latitude (beta)	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with <code>Longitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
Longitude (beta)	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Used with <code>Latitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
ManagerId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who manages this user.</p>
MobilePhone	<p>Type phone</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user's mobile or cellular phone number.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Concatenation of FirstName and LastName.</p>
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user's phone number.</p>
PostalCode	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user's postal or ZIP code. Label is Zip/Postal Code.</p>
SmallPhotoUrl	<p>Type url</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The URL for a thumbnail of the user's profile photo if Chatter is enabled. The URL is updated every time a photo is uploaded and reflects the most recent photo. The URL returned for an older photo is not guaranteed to return a photo if a newer photo has been uploaded. You should always query this field for the URL of the most recent photo.</p>
State	<p>Type string</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The state associated with the user profile.</p>
Street	<p>Type textarea</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The street address associated with the user profile.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user's business title, such as "Vice President."</p>
UserPreferencesActivityRemindersPopup	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, a reminder popup window automatically opens when an activity reminder is due. Corresponds to the <code>Trigger alert when reminder comes due</code> checkbox at the Reminders page in the personal settings in the user interface.</p>
UserPreferencesApexPagesDeveloperMode	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, indicates the user has enabled developer mode for editing Visualforce pages and controllers.</p>
UserPreferencesDisableAllFeedsEmail	<p>Type boolean</p> <p>Properties Filter</p>

Field	Details
UserPreferencesDisableBookmarkEmail	<p>Description When <code>false</code>, the user automatically receives email for all updates to Chatter feeds, based on the types of feed emails and digests the user has enabled.</p>
UserPreferencesDisableChangeCommentEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a Chatter feed item after the user has bookmarked it.</p>
UserPreferencesDisableEndorsementEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a change the user has made, such as an update to their profile.</p>
UserPreferencesDisableFeedbackEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the member automatically receives email every time someone endorses them for a topic.</p>
UserPreferencesDisableFileShareNotificationsForApi	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives emails related to Work.com feedback. This includes when someone requests or offers feedback, shares feedback with the user, or reminds the user to answer a feedback request.</p>

Field	Details
	<p>Properties Filter</p> <p>Description When <code>false</code>, email notifications are sent from the person who has shared a file to the users with whom the file has been shared.</p>
UserPreferencesDisableFollowersEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone starts following the user in Chatter.</p>
UserPreferencesDisableLaterCommentEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone comments on a feed item after the user has commented on the feed item.</p>
UserPreferencesDisableLikeEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone likes a post or comment the user has made.</p>
UserPreferencesDisableMentionsPostEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time the user is mentioned in posts.</p>
UserPreferencesDisableMessageEmail	<p>Type boolean</p>

Field	Details
	<p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email for Chatter messages sent to the user.</p>
UserPreferencesDisableProfilePostEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time someone posts to the user's profile.</p>
UserPreferencesDisableRewardEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives emails related to Work.com rewards. This includes when someone gives a reward to the user.</p>
UserPreferencesDisableSharePostEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user automatically receives email every time the user's post is shared.</p>
UserPreferencesDisableWorkEmail	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>false</code>, the user receives emails related to Work.com feedback, goals, and coaching. The user must also sign up for individual emails listed on the Work.com email settings page. When <code>true</code>, the user will not receive any emails related to Work.com feedback, goals, or coaching even if they are signed up for individual emails.</p>

Field	Details
UserPreferencesDisCommentAfterLikeEmail	Type boolean
	Properties Filter
	Description When <code>false</code> , the user automatically receives email every time someone comments on a post the user has liked.
UserPreferencesDisMentionsCommentEmail	Type boolean
	Properties Filter
	Description When <code>false</code> , the user automatically receives email every time the user is mentioned in comments.
UserPreferencesDisProfPostCommentEmail	Type boolean
	Properties Filter
	Description When <code>false</code> , the user automatically receives email every time someone comments on posts on the user's profile.
UserPreferencesEnableAutoSubForFeeds	Type boolean
	Properties Filter
	Description When <code>true</code> , the user automatically subscribes to feeds for any objects that the user creates.
UserPreferencesEventRemindersCheckboxDefault	Type boolean
	Properties Filter
	Description When <code>true</code> , a reminder popup is automatically set on the user's events. Corresponds to the <code>By default, set reminder on Events to...</code> checkbox on the Reminders page in the user interface. This field is related to UserPreference and customizing activity reminders.

Field	Details
UserPreferencesHideChatterOnboardingSplash	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the initial Chatter onboarding prompts do not appear.</p>
UserPreferencesHideCSNDesktopTask	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the Chatter recommendations panel never displays the recommendation to install Chatter Desktop.</p>
UserPreferencesHideCSNGetChatterMobileTask	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the Chatter recommendations panel never displays the recommendation to install Chatter Mobile.</p>
UserPreferencesHideS1BrowserUI	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Controls the interface that the user sees when logging in to Salesforce from a supported mobile browser. If <code>false</code>, the user is automatically redirected to the Salesforce1 mobile browser app. If <code>true</code>, the user sees the full Salesforce site. The default value is <code>false</code>. Label is Salesforce1 User. This field is available in API version 29.0 or later.</p>
UserPreferencesHideSecondChatterOnboardingSplash	<p>Type boolean</p> <p>Properties Filter</p> <p>Description When <code>true</code>, the secondary Chatter onboarding prompts do not appear.</p>

Field	Details
UserPreferencesReminderSoundOff	Type boolean
	Properties Filter
	Description <p>When <code>true</code>, a sound automatically plays when an activity reminder is due. Corresponds to the <code>Play a reminder sound</code> checkbox on the Reminders page in the user interface.</p>
UserPreferencesShowCityToExternalUsers	Type boolean
	Properties Filter
	Description <p>Indicates the visibility of the city field in the user's contact information. City is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none"> This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A. <p>City is visible to external members in a community when:</p> <ul style="list-style-type: none"> This field is <code>true</code>, or This field is <code>false</code> but <code>UserPreferencesShowCityToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowCityToGuestUsers	Type boolean
	Properties Filter
	Description <p>Indicates the visibility of the city field in the user's contact information. When <code>true</code>, city is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowCityToExternalUsers</code>, making the user's city visible to external members.</p>

Field	Details
UserPreferencesShowCountryToExternalUsers	<p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the country field in the user's contact information. Country is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none">• This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A. <p>Country is visible to external members in a community when:</p> <ul style="list-style-type: none">• This field is <code>true</code>, or• This field is <code>false</code> but <code>UserPreferencesShowCountryToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowCountryToGuestUsers	<p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p> <p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the country field in the user's contact information. When <code>true</code>, country is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p> <p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowCountryToExternalUsers</code>, making the user's country visible to external members.</p> <p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
UserPreferencesShowEmailToExternalUsers	<p>Type boolean</p>

Field	Details
	<p>Properties Filter</p> <p>Description Indicates the visibility of the email address field in the user's contact information. Email address is visible only to internal members of the user's organization when this field is <code>false</code>. Email address is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowFaxToExternalUsers	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the fax number field in the user's contact information. Fax number is visible only to internal members of the user's organization when this field is <code>false</code>. Fax number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowManagerToExternalUsers	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the manager field in the user's contact information. Manager is visible only to internal members of the user's organization when this field is <code>false</code>. Manager is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowMobilePhoneToExternalUsers	<p>Type boolean</p> <p>Properties Filter</p>

Field	Details
	<p>Description</p> <p>Indicates the visibility of the mobile or cellular phone number field in the user's contact information. The number is visible only to internal members of the user's organization when this field is <code>false</code>. The number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowPostalCodeToExternalUsers	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Filter</p> <p>Description</p> <p>Indicates the visibility of the postal or ZIP code field in the user's contact information. Postal code is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none">• This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A. <p>Postal code is visible to external members in a community when:</p> <ul style="list-style-type: none">• This field is <code>true</code>, or• This field is <code>false</code> but <code>UserPreferencesShowPostalCodeToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowPostalCodeToGuestUsers	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Filter</p> <p>Description</p> <p>Indicates the visibility of the postal or ZIP code field in the user's contact information. When <code>true</code>, postal code is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p>

Field	Details
	<p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowPostalCodeToExternalUsers</code>, making the user's postal code visible to external members.</p> <p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
<code>UserPreferencesShowProfilePicToGuestUsers</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the user's profile photo. When <code>true</code>, the photo is visible to guest users in a community. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community.</p> <p>When <code>false</code>, this field returns the stock photo. The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
<code>UserPreferencesShowStateToExternalUsers</code>	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the state field in the user's contact information. State is visible only to internal members of the user's organization when:</p> <ul style="list-style-type: none">• This field is <code>false</code>. When <code>false</code>, this field returns the value <code>#N/A</code>. <p>State is visible to external members in a community when:</p> <ul style="list-style-type: none">• This field is <code>true</code>, or• This field is <code>false</code> but <code>UserPreferencesShowStateToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>When <code>false</code>, this field returns the value <code>#N/A</code>. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
<code>UserPreferencesShowStateToGuestUsers</code>	<p>Type boolean</p>

Field	Details
	Properties
	Filter
	Description
	<p>Indicates the visibility of the state field in the user's contact information. When <code>true</code>, state is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A.</p>
	<p>When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowStateToExternalUsers</code>, making the user's state visible to external members.</p>
	<p>The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
<code>UserPreferencesShowStreetAddressToExternalUsers</code>	Type
	boolean
	Properties
	Filter
	Description
	<p>Indicates the visibility of the street address field in the user's contact information. The address is visible only to internal members of the user's organization when this field is <code>false</code>. The address is visible to external members in a community when this field is <code>true</code>.</p>
	<p>External users are users with Community, Customer Portal, or partner portal licenses.</p>
	<p>When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>
<code>UserPreferencesShowTitleToExternalUsers</code>	Type
	boolean
	Properties
	Filter
	Description
	<p>Indicates the visibility of the business title field in the user's contact information. Title is visible only to internal members of the user's organization when:</p>
	<ul style="list-style-type: none"> • This field is <code>false</code>. When <code>false</code>, this field returns the value #N/A.
	<p>Title is visible to external members in a community when:</p>
	<ul style="list-style-type: none"> • This field is <code>true</code>, or

Field	Details
UserPreferencesShowTitleToGuestUsers	<ul style="list-style-type: none">This field is <code>false</code> but <code>UserPreferencesShowTitleToGuestUsers</code> is <code>true</code>, which overrides this field's value. <p>External users are users with Community, Customer Portal, or partner portal licenses.</p> <p>The default value is <code>true</code>. This field is available in API version 26.0 and later.</p>
UserPreferencesShowWorkPhoneToExternalUsers	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the business title field in the user's contact information. When <code>true</code>, title is visible to guest users. Guest users can access public Site.com and Force.com sites, and public pages in Communities, via the Guest User license associated with each site or community. When <code>false</code>, this field returns the value #N/A. When <code>true</code>, this field overrides the value <code>false</code> in <code>UserPreferencesShowTitleToExternalUsers</code>, making the user's title visible to external members. The default value is <code>false</code>. This field is available in API version 28.0 and later.</p>
UserPreferencesTaskRemindersCheckboxDefault	<p>Type boolean</p> <p>Properties Filter</p> <p>Description Indicates the visibility of the work phone number field in the user's contact information. The number is visible only to internal members of the user's organization when this field is <code>false</code>. The number is visible to external members in a community when this field is <code>true</code>. External users are users with Community, Customer Portal, or partner portal licenses. When <code>false</code>, this field returns the value #N/A. The default value is <code>false</code>. This field is available in API version 26.0 and later.</p>

Field	Details
	Description
	When <code>true</code> , a reminder popup is automatically set on the user's tasks. Corresponds to the <code>By default, set reminder on Tasks to...</code> checkbox on the Reminders page in the user interface. This field is related to UserPreference and customizing activity reminders.

Usage

Use this object to query Chatter—related information about the user. While the User object contains all the information about a user and is historically tied to user management, UserProfile is a read-only entity that contains the information that is relevant in a Chatter context.

UserProfileFeed

Represents a user profile feed, which tracks all actions by a user on records that can be tracked in a feed. This feed is displayed on the user profile page.

UserProfileFeed is available in API version 18.0 through API 26.0. In API version 27.0 and later, UserProfileFeed is no longer available in the SOAP API. Use the Chatter REST API to access UserProfileFeed.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

- “Manage Unlisted Groups”

Only users with this permission can delete items in unlisted groups.

Fields

Field	Details
Body	Type
	textarea

Field	Details
	<p>Properties Nillable, Sort</p> <p>Description The content of UserProfileFeed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, CommentCount isn't updated until a comment is published. For example, say you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, your comment isn't counted until it's approved by an admin or a person with "CanApproveFeedPostAndComment" or "ModifyAllData" permission. This has implications for how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through CommentCount, go through pagination until end of comments is returned.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
	<p>Description This field is required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.</p>
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when this record was created. This field is a standard system field. Ordering by <code>CreatedDateDESC</code> sorts the feed by the most recent feed item.</p>
FeedPostId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in an FeedItem: changes to tracked fields, text posts, link posts, and content posts.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
	<p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the <code>InsertedBy</code> value is set to the ID of the logged-in user.</p>
IsDeleted	<p>Type <code>boolean</code></p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Standard system field. Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).</p>
LastModifiedDate	<p>Type <code>dateTime</code></p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user last modified this record. This field is a standard system field. When a feed item is created, <code>LastModifiedDate</code> is the same as <code>CreatedDate</code>. If a FeedComment is inserted on that feed item, then <code>LastModifiedDate</code> becomes the <code>CreatedDate</code> for that FeedComment. Deleting the FeedComment does not change the <code>LastModifiedDate</code>. Ordering by <code>LastModifiedDate DESC</code> sorts the feed by both the most recent feed item or comment.</p>
LikeCount	<p>Type <code>int</code></p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type <code>url</code></p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
NetworkScope	<p>Type <code>picklist</code></p>

Field	Details
	<p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none">• <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community.• <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none">• Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>.• For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>.• You can't filter a FeedItem on the <code>NetworkScope</code> field.
<code>ParentId</code>	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the account record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
<code>RelatedRecordId</code>	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>
<code>Title</code>	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the user profile record. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL and this field is the link name.</p>

Field	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item:</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • CaseCommentPost—generated event when a user adds a case comment for a case object

Field	Details
	<ul style="list-style-type: none"> • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received • <code>CallLogPost</code>—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • <code>ChangeStatusPost</code>—generated event when a user changes the status of a case • <code>AttachArticleEvent</code>—generated event when a user attaches an article to a case <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>
<code>Visibility</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if Salesforce Communities is enabled for your organization.</p> <p><code>Visibility</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>AllUsers</code>—The feed item is available to all users who have permission to see the feed item. • <code>InternalUsers</code>—The feed item is available to internal users only. <p>Note the following exceptions for <code>Visibility</code>:</p> <ul style="list-style-type: none"> • For record posts, <code>Visibility</code> is set to <code>InternalUsers</code> for all internal users by default. • External users can set <code>Visibility</code> only to <code>AllUsers</code>. • On user and group posts, only internal users can set <code>Visibility</code> to <code>InternalUsers</code>.

Usage

Use this object to query and retrieve record changes tracked in a user profile feed. Note the following when working with user profile feeds:

- This object is read only in the API.
- Queries retrieve feed items that include mentions from other users.
- Use this object to query and retrieve lead feed items that were associated with a converted lead record.
- Include a `WITH` clause and specify the `UserId` of the user whose profile you want to query. The `WITH` clause must come after a `WHERE` clause.

Users that do not have the "View All Data" permission have the following limitations when querying records:

- Must specify a `LIMIT` clause and the limit must be less than or equal to 1000.

- Can include a WHERE clause that references UserProfileFeed fields, but cannot include references to fields in related objects. For example, you can filter by CreatedDate or Parent.Id, but not by Parent.Name.
- Can include an ORDER BY clause that references UserProfileFeed fields, but cannot include references to fields in related objects. For example, you can ORDER BY CreatedDate or Parent.Id, but not by Parent.Name.

 **Tip:** To query for the most recent feed items, you should ORDER BY CreatedDate DESC, Id DESC.

Note the following SOQL restrictions.

- No SOQL limit if logged-in user has “View All Data” permission. If not, specify a LIMIT clause of 1,000 records or fewer.
- SOQL ORDER BY on fields using relationships is not available. Use ORDER BY on fields on the root object in the SOQL query.

SEE ALSO:

[EntitySubscription](#)
[FeedComment](#)
[FeedTrackedChange](#)

UserProvAccount

Represents information that links a Salesforce user account with an account in a third-party (target) system, such as Google, for users of connected apps with Salesforce user provisioning enabled. This object is available in API version 33.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field	Details
ConnectedAppId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The 15 character application ID.</p>
DeletedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date and time when the associated user account in the target system was deleted. This value is automatically updated during the provisioning and reconciling processes.</p>

Field	Details
ExternalEmail	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The email address as stored in the target system for the associated user account.</p>
ExternalFirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The first name as stored in the target system for the associated user account.</p>
ExternalLastName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The last name as stored in the target system for the associated user account.</p>
ExternalUserId	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p>Description The unique identifier for the user as stored in the target system.</p>
ExternalUsername	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The username as stored in the target system for the associated user account.</p>
IsKnownLink	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description</p> <p>Setting the <code>IsKnownLink</code> value to <code>true</code> implies the administrator or another user is managing the relationship between the Salesforce user account and the third-party user account, manually. This field helps Salesforce coordinate updates between the <code>UserProvAccountStaging</code> object and the <code>UserProvAccount</code> object while committing staged accounts. Typically, for a matching user account (the same <code>ExternalUserId</code> for both objects), Salesforce copies the values from the <code>UserProvAccountStaging</code> object to the <code>UserProvAccount</code> object.</p> <p>However, if Salesforce encounters a <code>UserProvAccountStaging</code> object with a matching <code>ExternalUserId</code> but different <code>LinkState</code> and <code>SalesforceUserId</code> values during this process, Salesforce checks the <code>UserProvAccount</code> <code>IsKnownLink</code> value. If the <code>IsKnownLink</code> value is <code>true</code>, Salesforce doesn't copy the <code>LinkState</code> and <code>SalesforceUserId</code> values from the <code>UserProvAccountStaging</code> object to the <code>UserProvAccount</code> object (all other values are copied).</p> <p>The default is <code>false</code>, meaning Salesforce manages the account relationship.</p>
LinkState	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description</p> <p>The state of the current connection between the user account in the Salesforce organization and the associated user account in the target system. The valid values are:</p> <ul style="list-style-type: none"> • <code>linked</code>— changes to the account in the Salesforce organization are queued to be updated for the associated user account in the target system. • <code>duplicate</code>— an associated account in the target system exists. • <code>orphaned</code>—no associated account exists in the target system. • <code>ignored</code>— changes to the account in the Salesforce organization have no effect on the associated user account in the target system.
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description</p> <p>The unique name for this object.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Namepointing, Sort, Update</p>

Field	Details
	<p>Description The user ID of the owner of this object—typically a Salesforce administrator.</p>
SalesforceUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user ID for the user account in the Salesforce organization that is associated with the user account in the target system.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The status of the account in the target system. The valid values are:</p> <ul style="list-style-type: none"> • Active • Deactivated • Deleted

UserProvAccountStaging

Temporarily stores user account information while a user completes the User Provisioning Wizard. This information that is stored in the UserProvAccount object when you click the button to collect and analyze accounts on the target system.

User provisioning links a Salesforce user account with an account in a third-party (target) system. To configure user provisioning, you use a User Provisioning Wizard that guides you through the setup process. As you enter values about account details in the wizard, these values are stored in this object until you click the button to collect and analyze accounts on the target system. The general user provisioning configuration details are stored in the UserProvisioningConfig object.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
ConnectedAppId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The 15 character connected app ID.</p>
ExternalEmail	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The email address as stored in the target system for the associated user account.</p>
ExternalFirstName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The first name as stored in the target system for the associated user account.</p>
ExternalLastName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The last name as stored in the target system for the associated user account.</p>
ExternalUserId	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p>Description The unique identifier for the user as stored in the target system.</p>
ExternalUsername	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The username as stored in the target system for the associated user account.</p>

Field Name	Details
LinkState	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The state of the current connection between the user account in the Salesforce organization and the associated user account in the target system. The valid values are:</p> <ul style="list-style-type: none"> • linked— a user account matches one in the target system. • duplicate— an associated account in the target system exists. • orphaned—no associated account exists in the target system. • ignored— changes to the account in the Salesforce organization have no effect on the associated user account in the target system.
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The unique name for this object.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The user ID of the owner of this object—typically a Salesforce administrator.</p>
SalesforceUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The user ID for the user account in the Salesforce organization that is associated with the user account in the target system.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	Description
	<p>The status of the account in the target system. The valid values are:</p> <ul style="list-style-type: none"> • Active • Deactivated • Deleted

Usage

When committing fields from a UserProvAccountStaging to a UserProvAccount object, Salesforce looks up the UserProvAccount record where `UserProvAccountStaging.ExternalUserId = UserProvAccount.ExternalUserId`.

- If an `ExternalUserId` doesn't match an existing account, Salesforce creates a UserProvAccount record based on the UserProvAccountStaging record.
- If an `ExternalUserId` matches, then Salesforce checks the `UserProvAccount.isKnownLink` value, and does the following.
 - If `UserProvAccount.IsKnownLink = true`, Salesforce copies the UserProvAccountStaging values to the UserProvAccount object, except for the `ExternalUserId` and `LinkState` values.
 - If `UserProvAccount.IsKnownLink = false`, Salesforce copies all of the UserProvAccountStaging values to the UserProvAccount object.

UserProvMockTarget

Represents an entity for testing user data before committing the data to a third-party system for user provisioning.

During the user provisioning process, user account information is sent to a third-party system to create, update or delete a user account on that system. While configuring user provisioning for your organization using a flow or Apex plugin, you can use this object to confirm the associated flow or plugin is sending the desired data. After confirming the correct fields and values, you can update the flow or Apex plugin to send the data to the target system.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field Name	Details
	Type
<code>ExternalEmail</code>	<p>string</p> <p>Properties</p> <p>Create, Filter, Group, Nullable, Sort, Update</p>

Field Name	Details
Description	
ExternalFirstName	<p>The email address as stored in the target system for the associated user account.</p>
Type	
ExternalFirstName	string
Properties	
ExternalFirstName	Create, Filter, Group, Nillable, Sort, Update
Description	
ExternalFirstName	The first name as stored in the target system for the associated user account.
ExternalLastName	
Type	
ExternalLastName	string
Properties	
ExternalLastName	Create, Filter, Group, Nillable, Sort, Update
Description	
ExternalLastName	The last name as stored in the target system for the associated user account.
ExternalUserId	
Type	
ExternalUserId	string
Properties	
ExternalUserId	Create, Filter, Group, idLookup, Nillable, Sort, Update
Description	
ExternalUserId	The unique identifier for the user as stored in the target system.
ExternalUsername	
Type	
ExternalUsername	string
Properties	
ExternalUsername	Create, Filter, Group, Nillable, Sort, Update
Description	
ExternalUsername	The username as stored in the target system for the associated user account.
Name	
Type	
Name	string
Properties	
Name	Create, Filter, Group, idLookup, Sort, Update
Description	
Name	The unique name for this object.
OwnerId	
Type	
OwnerId	reference

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>
	<p>Description The user ID of the owner of this object—typically a Salesforce administrator.</p>

UserProvisioningConfig

Represents information for a flow to use during a user provisioning request process, such as the attributes for an update. This object is available in API version 34.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
ApprovalRequired	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Denotes whether approvals are required for provisioning users for the associated connected app. If the value is null, no approval is required.</p>
ConnectedAppId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The 18-digit application ID for the connected app.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with</p>

Field Name	Details
	<p>a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package, and the changes are reflected in a subscriber's organization.</p>
Enabled	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether user provisioning is enabled for the associated connected app (<code>true</code>) or not (<code>false</code>).</p>
EnabledOperations	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Lists the operations, as comma-separated values, that create a UserProvisioningRequest object for the associated connected app. Allowed values are:</p> <ul style="list-style-type: none">• Create• Update• EnableAndDisable (activation and deactivation)• SuspendAndRestore (freeze and unfreeze)
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The two- to five-character code that represents the language and locale ISO. This code controls the language for labels displayed in an application.</p>
LastReconDateTime	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>The date and time when user accounts were last reconciled between Salesforce and the target system.</p>
MasterLabel	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Sort, Update</p> <p>Description</p> <p>The master label for this object. This value is the internal label that doesn't get translated.</p>
NamedCredentialId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>Salesforce ID of the named credential that's used for a request. The named credential identifies the third-party system and the third-party authentication settings.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none">• In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.• In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Field Name	Details
Notes	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description A utility field for administrators to add any additional information about the configuration. This field is for internal reference only, and is not used by any process.</p>
OnUpdateAttributes	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Lists the user attributes, as comma-separated values, that generate a UserProvisioningRequest object during an update.</p>
ReconFilter	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description When collecting and analyzing users on a third-party system, the plug-in uses this filter to limit the scope of the collection.</p>
UserAccountMapping	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Stores the attributes used to link the Salesforce user to the account on the third-party system, in JSON format. For example: <pre>{"linkingSalesforceUserAttribute": "Username", "linkingTargetUserAttribute": "Email"}</pre></p>

UserProvisioningLog

Represents messages generated during the process of provisioning users for third-party applications. This object is available in API version 33.0 and later.

Some messages for this object are generated automatically by Salesforce, and others are created by the developers of the user provisioning plugin. Developers can use this object to log messages from the flow associated with the user provisioning process or the Apex plugin that calls the target system. Administrators can use this object as a log of all user provisioning activity and as a troubleshooting tool if desired behavior is missing. This object is available as a custom report type.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Fields

Field	Details
Details	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The value of this field depends on the log entry. For example, if the target system returns an error, the error message may be recorded in this field.</p>
ExternalUserId	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique identifier for the user in the target system.</p>
ExternalUsername	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The username set in the target system for the associated user account.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p>

Field	Details
	<p>Description The unique name for this object.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Salesforce ID of the Group or User who owns this object.</p>
Status	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The status of the user provisioning request. Based on the context of the log, it can contain different values, such as an HttpStatusCode.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Salesforce ID of the user making the request.</p>
UserProvisioningRequestId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A unique identifier for the user provisioning request.</p>

UserProvisioningRequest

Represents an individual provisioning request to create, update, or delete a single user account in a third-party service system (or another Salesforce organization). This object is available in API version 33.0 and later.

A UserProvisioningRequest (UPR) record is created for each provisioning action for each user, and for each connected app available to the user. For example, if a user has two connected apps, and a provisioning request is sent to two different services to create an account for the user, Salesforce creates two UPR objects. Provisioning actions include creating, updating, or deleting a user account.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field	Details
AppName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the connected app associated with the service provider.</p>
ApprovalStatus	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status of the approval for the current request. If the user provisioning setup for the connected app does not have an approval process enabled, the status is Not Required. If an approval process is enabled, supported values are:</p> <ul style="list-style-type: none"> • Required— An approval process is enabled in the user provisioning setup for the associated connected app, but there is no response to the request yet. • Not Required— An approval process is not enabled in the user provisioning setup for the associated connected app. • Approved • Denied
ConnectedAppId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The 18-digit application ID for the connected app.</p>
ExternalUserId	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique identifier for the user in the target system.</p>

Field	Details
ManagerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Salesforce ID of the user who manages the user specified in the <code>SalesforceUserId</code> field. If an approval process is configured for the user provisioning request, this value allows the manager to approve the request. Available in API version 34.0 and later.</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The unique name for this object.</p>
Operation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The Apex method called by the trigger associated with the provisioning request (typically a change to the User object). Supported values are:</p> <ul style="list-style-type: none"> • Create • Read • Update • Deactivate • Activate • Freeze • Unfreeze • Reconcile • Linking <p>For example, when the User object field <code>isActive</code> is set to <code>false</code>, the UPR object <code>Operation</code> field value is set to <code>Deactivate</code>.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description Salesforce ID of the Group or User who owns this object.</p>
ParentID	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description When a retry event is created, the failed UPR is cloned and resubmitted. This field contains a lookup to the failed UPR that was cloned to create the current record.</p>
Retry Count	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Number of retry attempts performed on a UPR. Retry Count enables custom business logic such as "Retry 5 times then stop and notify your admin."</p>
SalesforceUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Salesforce ID of the user making the request.</p>
ScheduleDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description When to send this request to the service provider.</p> <p> Note: Scheduling is not implemented yet. Currently, provisioning changes are queued immediately to be sent to the service provider.</p>
State	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Status of this request. Supported values are:</p>

Field	Details
	<ul style="list-style-type: none"> • New • Requested • Completed • Failed • Collecting • Collected • Analyzing • Analyzed • Committing • Retried • Manually Completed
	<p>The State goes from New to Requested to Completed or Failed, unless a reconciliation process is occurring. For details about the reconciliation process State value changes, see Usage.</p>
	<p>The State goes from Failed to Retried or Manually Completed when troubleshooting UPR failures. For details about handling failures, see State Values for Managing Provisioning Failures.</p>
UserProvAccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID value of the associated UserProvAccount object.</p>
UserProvConfigId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID value of the associated UserProvisioningConfig object. Available in API version 34.0 and later.</p>

Usage

The State value changes during a reconciliation process (Operation = Reconcile) to gather and compare users on the third-party system to Salesforce users. Typically, when a UPR entry is first created, it has a State value of New. When a collection process is triggered, the State transitions to Collecting until that process is finished and the State is Collected. When an analyze process is triggered, the State transitions to Analyzing until that process is finished and the State is Analyzed. If a process commits the request, the State then transitions to Committing, and the properties move from the

UserProvAccountStaging object to the UserProvAccount object. When those properties are saved in the UserProvAccount object, the State transitions to **Completed**.

However, the State does not necessarily start at **New**. For example, UserProvAccountStaging entries can be inserted programmatically. If a process is initiated that triggers linking these rows to accounts on the third-party service, a UPR entry could start with the **Analyzing** State.

Also, the State cannot go backwards from an active task. For example, a successful **Analyzing** State must progress to **Analyzed**; unless the active process fails, and then the State must change to **Failed**. Certain State transitions cannot be made programmatically and must be triggered by Salesforce.

The following table shows the State transitions that can occur for each State value. Each row corresponds to a current State value and each column corresponds to a new State after a potential transition.

- — the transition to this value is not allowed.
- — the transition to this value is allowed.
- — only Salesforce can transition the State to this value.

	New	Requested	Collecting	Collected	Analyzing	Analyzed	Committing	Completed	Failed	Retried	Manually Completed
New											
Requested											
Collecting											
Collected											
Analyzing											
Analyzed											
Committing											
Completed											
Failed											
Retried											
Manually Completed											

State Values for Managing Provisioning Failures

The state value changes to **Failed** for several reasons, such as network outages, session timeouts, permissions issues, and record locks. The **Failed** state can transition to either **Retried** or **Manually Completed** to indicate what action was taken to address the failure. Actions can include correcting the root cause of the failure and requesting that the provisioning engine retry the UPR. Or, it can be completing the action against the target manually. Each UPR is an independent transaction and it's possible the retry causes a failure with a different root cause. So it's hard to distinguish failed events that you addressed from the ones that require more action.

If you tried to correct the cause of the failure and requested the provisioning engine to retry the UPR, you can mark the failed UPR **Retried**. Or, if the action against the target was completed manually, you can mark it **Manually Completed**.

When a retry event is created, the failed UPR is cloned, and resubmitted. The `ParentID` field contains a lookup to the failed UPR to use to clone the new UPR. The `Retry Count` field contains the number of retry attempts that were performed on a UPR. With the `Retry Count` field, you can add custom business logic like "Retry 5 times then stop and notify your admin."

UserProvisioningRequestOwnerSharingRule

Represents a rule for sharing a UserProvisioningRequest object with users other than the owner. This object is available in API version 34.0 and later.

 **Note:** To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`update()`,`upsert()`

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing allowed. The possible values are:</p> <ul style="list-style-type: none">• Read• Edit
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description A description of the sharing rule. Maximum size is 1000 characters.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two</p>

Field Name	Details
	consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
GroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the source group. UserProvisioningRequest objects owned by users in the source group trigger the rule to give access.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to Label in the user interface.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID representing the target user or group. Target users or groups have access to the UserProvisioningRequest object.</p>

SEE ALSO:

[Metadata API Developer Guide: SharingRules](#)

UserProvisioningRequestShare

Represents a sharing entry on a UserProvisioningRequest record. This object is available in API version 34.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description A value that represents the type of sharing allowed. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the parent record, if any.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Values may include:</p> <ul style="list-style-type: none"> • Manual—The User or Group has access because a user with “All” access manually shared the UserProvisioningRequest record with them. • Owner—The User is the owner of the UserProvisioningRequest record or is in a role above the UserProvisioningRequest record owner in the role hierarchy.
UserOrGroupId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort</p>
	<p>Description ID of the User or Group that has been given access to the UserProvisioningRequest record.</p>

UserRecordAccess

Represents a user's access to a set of records. This object is read only and is available in API version 24.0 and later.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field	Details
HasAllAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a user can share the record.</p>
HasDeleteAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a user has delete access to the record (<code>true</code>) or not (<code>false</code>).</p>
HasEditAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a user has edit access to the record (<code>true</code>) or not (<code>false</code>).</p>

Field	Details
HasTransferAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a user has transfer access to the record (<code>true</code>) or not (<code>false</code>).</p>
HasReadAccess	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a user has read access to the record (<code>true</code>) or not (<code>false</code>).</p>
MaxAccessLevel	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p> <p>Description Indicates a user's maximum level of access to a record.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • None • Read • Edit • Delete • Transfer • All
RecordId	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description ID of the record.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group</p>

Field	Details
Description	ID of the user.

Usage

Use this object to query a user's access to records. You can't create, delete, or update any records using this object.

Up to 200 record IDs can be queried. You can include an `ORDER BY` clause for any field that is being selected in the query.

The following sample query returns the records, whether the queried user has read and transfer access to each record, and the user's maximum access level to each record.

```
SELECT RecordId, HasReadAccess, HasTransferAccess, MaxAccessLevel
    FROM UserRecordAccess
    WHERE UserId = [single ID]
    AND RecordId = [single ID]      //or Record IN [list of IDs]
```

The following query returns the records to which a queried user has read access.

```
SELECT RecordId
    FROM UserRecordAccess
    WHERE UserId = [single ID]
    AND RecordId = [single ID]      //or Record IN [list of IDs]
    AND HasReadAccess = true
```

Using API version 30.0 and later, `UserRecordAccess` is a foreign key on the records. You can't filter by or provide the `UserId` or `RecordId` fields when using this object as a lookup or foreign key. The previous sample queries can be run as:

```
SELECT Id, Name, UserRecordAccess.HasReadAccess, UserRecordAccess.HasTransferAccess,
UserRecordAccess.MaxAccessLevel
    FROM Account
```

```
SELECT Id, Name, UserRecordAccess.HasReadAccess
    FROM Account
```

SOQL restrictions for API version 29.0 and earlier:

- When the running user is querying a user's access to a set of records, records that the running user does not have read access to are filtered out of the results.
- When filtering by `UserId` and `RecordId` only, you must use `SELECT RecordId` and optionally one or more of the access level fields: `HasReadAccess`, `HasEditAccess`, `HasDeleteAccess`, `HasTransferAccess`, and `HasAllAccess`. You may include `MaxAccessLevel`.
- When filtering by `UserId`, `RecordId`, and an access level field, you must use `SELECT RecordId` only.

UserRole

Represents a user role in your organization.

 **Note:** This object was called "Role" in previous versions of the API documentation.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
CaseAccessForAccountOwner	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The case access level for the account owner.</p>
ContactAccessForAccountOwner	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The contact access level for the account owner.</p> <p> Note: When DefaultContactAccess is set to Controlled by Parent, you can't create or update this field.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to Role Name in the user interface.</p> <p>This field is available in API version 24.0 and later.</p>

Field	Details
	<p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
ForecastUserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the forecast manager associated with this role. Label is User ID.</p>
IsPartner	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the user role is a partner who has access to the partner portal (<code>true</code>) or not (<code>false</code>). This field is not available for release 9.0 and later. Instead, use <code>PortalType</code> with the value <code>Partner</code>.</p>
MayForecastManagerShare	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the forecast manager can manually share their own forecast.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name of the role. Corresponds to Label on the user interface.</p>
OpportunityAccessForAccountOwner	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field	Details
	<p>Description Required. The opportunity access level for the account owner. Note that you can't set a user role with an opportunity access less than that specified in organization-wide defaults.</p>
ParentRoleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the parent role.</p>
PortalRole	<p>Type picklist</p> <p>Properties Filter, Nillable</p> <p>Description The portal role: Executive, Manager, User, or PersonAccount.</p>
PortalType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description This value indicates the type of portal for the role:</p> <ul style="list-style-type: none">• None: Salesforce application role.• CustomerPortal: Customer portal role.• Partner: partner portal role. The field <code>IsPartner</code> used in release 8.0 will map to this value. <p>This field replaces <code>IsPartner</code> beginning with release 9.0.</p>
RollupDescription	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Description of the forecast rollup. Label is Description.</p>

Usage

Use this object to query the set of currently configured user roles in your organization. Use it in your client application to obtain valid UserRole IDs to use when querying or modifying a User record.

All users have access to invoke query or describe this object. If your client application logs in with the “Manage Users” permission, it can query, create, update, or delete UserRole records.

 **Note:** You can't update any field for a portal role.

For example, the following code finds all roles that are not assigned to any users.

```
SELECT Id, Name, DeveloperName,  
FROM UserRole  
WHERE Id NOT IN (SELECT UserRoleId  
                  FROM User  
                  WHERE UserRoleId !='0000000000000000')
```

SEE ALSO:

[Object Basics](#)

UserServicePresence

Represents a presence user’s real-time presence status. This object is available in API version 32.0 and later.

Supported Calls

`query(), getDeleted(), getUpdated(), retrieve()`

Fields

Field	Details
Capacity	<p>Type double</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The user’s capacity to take on work items.</p>
ConfiguredCapacity	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user’s total configured capacity.</p>

Field	Details
IsAway	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the user's status is <code>Away</code> .
Name	Type string Properties Create, Filter, Nullable, Sort, Update Description An automatically generated ID number that identifies the record.
ServicePresenceStatusId	Type reference Properties Create, Filter, Nullable, Sort, Update Description The ID of the presence status that's associated with the presence user that's specified by the <code>UserId</code> .
UserId	Type string Properties Create, Filter, Group, Sort, Update Description The ID of the Omni-Channel user.

UserShare

Represents a sharing entry on a user record. This object is available in API version 26.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Read-only. Indicates whether the User has access to log in (<code>true</code>) or not (<code>false</code>). You can modify a User's active status from the user interface or via the API.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. You can only write to this field when its value is either omitted or set to <code>Manual</code> (default). You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Possible values include:</p> <ul style="list-style-type: none"> • <code>Manual</code>—The User or Group has access to the user record because a User with “All” access manually shared the User with them. • <code>Rule</code>—The User or Group has access to the user record via a User sharing rule.
UserAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Level of access that the User or Group has to the specified user. The specified user is denoted by the <code>UserId</code>. The possible values are:</p> <ul style="list-style-type: none"> • <code>Read</code> • <code>Edit</code> <p>This field must be set to an access level that is at least equal to the organization’s default <code>UserAccessLevel</code>. <code>UserAccessLevel</code> can be updated only if <code>RowCause</code> is set to <code>Manual Sharing</code>.</p>

Field	Details
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User being shared.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the User or Group that has been given access to the User. This field can't be updated.</p>

Usage

This object allows you to determine which users and groups can view or edit User records owned by other users.

UserTeamMember

Represents a single User on the default opportunity team of another User.

Supported Calls

```
create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(),
upsert()
```

Special Access Rules

- This object is available only in organizations that have enabled the team selling functionality.
- Customer Portal users can't access this object.

Fields

Field	Details
OpportunityAccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Restricted picklist, Update</p>

Field	Details
	<p>Description</p> <p>Required. Level of access that the team member has to opportunities for which the user has added his or her default opportunity team. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit <p>This field must be set to an access level that is higher than the organization's default access level for opportunities.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description</p> <p>Required. ID of the User who owns the default opportunity team. This field can't be updated.</p>
TeamMemberRole	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Update</p> <p>Description</p> <p>Role that the team member has on opportunities for which the User has added his or her default opportunity team. The valid values are set by the organization's administrator in the Opportunity Team Roles picklist. Label is Team Role.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description</p> <p>Required. ID of the User who is a member of the default opportunity team. This field can't be updated.</p>

Usage

If you attempt to create a record that matches an existing record, the create request updates any modified fields and returns the existing record.

Users can set up their default opportunity team to include other users that typically work with them on opportunities.

SEE ALSO:

[OpportunityTeamMember](#)

UserTerritory

Represents a User who has been assigned to a Territory.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

- Only available if territory management has been enabled for your organization.
- Customer Portal users can't access this object.

Fields

Field	Details
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the user is active in the given territory (<code>true</code>), or inactive in the given territory (<code>false</code>):</p> <ul style="list-style-type: none">Users who are active in a territory are explicitly assigned to the territory and can have open opportunities, closed opportunities, or no opportunities associated with that territory.Users who are inactive in a territory are not explicitly assigned to the territory, but own an open or closed opportunity that is associated with the territory. For example, a user may have been transferred out of a territory, but still own opportunities in his or her old territory. <p>Until a user is deleted from a territory (not simply removed from the territory), the record is not returned in a <code>getDeleted()</code> call.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.</p>
TerritoryId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter</p> <p>Description ID of the Territory to which the user has been assigned. This field is required when creating a record in API version 20.0 and later.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter</p> <p>Description ID of the user. This field is required when creating a record.</p>

Usage

If a user is inactive in a territory, and the opportunities they own that are associated with the territory are all closed, then the user is not returned.

SEE ALSO:

[Territory](#)
[AccountTerritoryAssignmentRule](#)
[AccountTerritoryAssignmentRuleItem](#)

UserTerritory2Association

Represents an association (by assignment) between a territory and a user record. Available only if Enterprise Territory Management has been enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve()`

Special Access Rules

If a territory model is in `Active` state, any user can view that model, including its territories and assignment rules. For territories in an active model, any user can view assigned records and assigned users subject to your organization's sharing settings. Users cannot view territory models in other states (such as `Planning` or `Archived`).

Fields

Field Name	Details
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the user is active (<code>true</code>) or inactive (<code>false</code>) in the given territory.</p>
RoleInTerritory2	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The role of the user in a territory. Possible values are: Owner, Administrator, Sales Rep. Label is <code>Role in Territory</code>.</p>
Territory2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the territory that the user is assigned to.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user who is assigned to the territory.</p>

VerificationHistory

Represents the past six months of your org users' attempts to verify their identity. This object is available in API version 36.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Only users with “Manage Users” permission can access this object.

Fields

Field Name	Details
Activity	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The action the user attempted that requires identity verification. The label is User Activity. Available values are:</p> <ul style="list-style-type: none"> • AccessReports—The user attempted to access reports or dashboards. • Apex—The user attempted to access a Salesforce resource with a verification Apex method. • ConnectedApp—The user attempted to access a connected app. • EnableLL—The user attempted to enroll in Lightning Login. This value is available in API version 38.0 and later. • ExportPrintReports—The user attempted to export or print reports or dashboards. • Login—The user attempted to log in. • Registration—Reserved for future use. • TempCode—The user attempted to generate a temporary verification code. This value is available in API version 37.0 and later.
EventGroup	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the verification attempt. Verification can involve several attempts and use different verification methods. For example, in a user’s session, a user enters an invalid verification code (first attempt). The user then enters the correct code and successfully verifies identity (second attempt). Both attempts are part of a single verification and, therefore, have the same ID. The label is Verification Attempt.</p>
LoginGeoId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field Name	Details
LoginHistoryId	Description The 18-character ID for the record of the geographic location of the user for a successful or unsuccessful identity verification attempt. Due to the nature of geolocation technology, the accuracy of geolocation fields (for example, country, city, postal code) may vary.
Policy	Type reference
Policy	Properties Filter, Group, Sort
Policy	Description The ID for the record of the user's successful or unsuccessful login attempt.
Policy	Type picklist
Policy	Properties Filter, Group, Restricted picklist, Sort
Policy	Description The identity verification security policy or setting. The label is Triggered By. Available values are:
	<ul style="list-style-type: none"> • CustomApex—Identity verification made by a verification Apex method. • DeviceActivation—Identity verification required for users logging in from an unrecognized device or new IP address. This verification is part of Salesforce's risk-based authentication. • EnableLightningLogin—Identity verification required for users enrolling in Lightning Login. This verification is triggered when the user attempts to enroll. Users are eligible to enroll if they have the "Lightning Login User" user permission and the org has enabled "Allow Lightning Login" in Session Settings. • HighAssurance—High assurance session required for resource access. This verification is triggered when the user tries to access a resource, such as a connected app, report, or dashboard that requires a high-assurance session level. • LightningLogin—Identity verification required for users logging in via Lightning Login. This verification is triggered when the enrolled user attempts to log in. Users are eligible to log in if they have the "Lightning Login User" user permission, have successfully enrolled in Lightning Login, and the org has enabled "Allow Lightning Login" in Session Settings. • ProfilePolicy—Session security level required at login. This verification is triggered by the "Session security level required at login" setting on the user's profile. • TwoFactorAuthentication—Two-factor authentication required at login. This verification is triggered by the "Two-Factor Authentication for User Interface

Field Name	Details
Remarks	<p>Logins" user permission assigned to a custom profile. Or, the user permission is included in a permission set that is assigned to a user.</p>
ResourceId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the <code>Activity</code> value is <code>ConnectedApp</code>, the <code>ResourceId</code> value is the ID of the connected app. The label is <code>Connected App ID</code>.</p>
SourceIp	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The IP address of the machine from which the user attempted the action that requires identity verification. For example, the IP address of the machine from where the user tried to log in or access reports. If it's a non-login action that required verification, the IP address can be different from the address from where the user logged in. This address can be an IPv4 or IPv6 address.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The status of the identity verification attempt. Available values are:</p>

Field Name	Details
	<ul style="list-style-type: none"> AutomatedSuccess—Salesforce Authenticator approved the request for access because the request came from a trusted location. After users enable location services in Salesforce Authenticator, they can designate trusted locations. When a user trusts a location for a particular activity, such as logging in from a recognized device, that activity is approved from the trusted location for as long as the location is trusted. Denied—The user denied the approval request in the authenticator app, such as Salesforce Authenticator. FailedGeneralError—An error caused by something other than an invalid verification code, too many verification attempts, or authenticator app connectivity. FailedInvalidCode—The user provided an invalid verification code. FailedTooManyAttempts—The user attempted to verify identity too many times. For example, the user entered an invalid verification code repeatedly. Initiated—Salesforce initiated identity verification but hasn't yet challenged the user. InProgress—Salesforce challenged the user to verify identity and is waiting for the user to respond or for Salesforce Authenticator to send an automated response. RecoverableError—Salesforce can't reach the authenticator app to verify identity, but will retry. ReportedDenied—The user denied the approval request in the authenticator app, such as Salesforce Authenticator, and also flagged the approval request to report to an administrator. Succeeded—The user's identity was verified.
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the user verifying identity.</p>
VerificationMethod	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The method by which the user attempted to verify identity in the verification event. The label is Method. Available values are:</p> <ul style="list-style-type: none"> Email—Salesforce sent an email with a verification code to the address associated with the user's account.

Field Name	Details
	<ul style="list-style-type: none"> EnableLL—Salesforce Authenticator sent a notification to the user’s mobile device to enroll in Lightning Login. This value is available in API version 38.0 and later. Hotp—Reserved for future use. LL—Salesforce Authenticator sent a notification to the user’s mobile device to approve login via Lightning Login. This value is available in API version 38.0 and later. SalesforceAuthenticator—Salesforce Authenticator sent a notification to the user’s mobile device to verify account activity. Sms—Salesforce sent a text message with a verification code to the user’s mobile device. TempCode—A Salesforce admin or a user with the “Manage Two-Factor Authentication in User Interface” permission generated a temporary verification code for the user. This value is available in API version 37.0 and later. Totp—An authenticator app generated a time-based, one-time password (TOTP) on the user’s mobile device. U2F—A U2F security key generated required credentials for the user. This value is available in API version 38.0 and later.
VerificationTime	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The time of the identity verification attempt. The time zone is based on GMT. The label is Time.</p>

Usage

Here are two examples of the types of API queries you can perform.

Query	String
Show verification history for a user’s login record	<pre>SELECT Activity, EventGroup, Policy, Remarks, Status, UserId, VerificationMethod, VerificationTime FROM VerificationHistory WHERE LoginHistoryId = '0Yad000#####'</pre>
Get detailed geographic location information for a user’s verification attempt	<pre>SELECT City, CountryIso, Latitude, Longitude, PostalCode FROM LoginGeo WHERE LoginGeoId = '0LE#####'</pre>

Vote

Represents a vote that a user has made on an Idea or a Reply.

 **Note:** In API version 16.0 and earlier, SOQL queries on the Vote object only return votes for the Idea object. Starting in API version 17.0, SOQL queries return votes for both Idea and Reply.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates whether the object has been moved to the Recycle Bin (true) or not (false). Label is Deleted.</p>
ParentId	<p>Type reference</p> <p>Properties Group, Sort, Create, Filter</p> <p>Description ID of the Idea or Reply associated with this vote.</p>
Type	<p>Type picklist</p> <p>Properties Group, Sort, Create, Filter, Restricted picklist</p> <p>Description Picklist that indicates the type of vote. The value <code>Up</code> indicates that the vote is a user's positive endorsement of the associated idea or reply. The value <code>Down</code> indicates that the vote is a user's negative endorsement of the associated idea or reply.</p>

 **Note:** If you are importing Vote data into Salesforce and need to set the value for an audit field, such as `CreatedDate`, contact Salesforce. Audit fields are automatically updated during API operations unless you request to set these fields yourself.

Usage

In version 12.0 and later, use this object to track the votes that users made on ideas. For more information on ideas, see “Ideas Overview” in the Salesforce online help.

In version 17.0 and later, use this object to track the votes users made on replies. For more information, see “Answers Overview” in the Salesforce online help.

In version 17.0 and later, you must filter using the following syntax when querying this object in a SOQL query: `ParentId = single ID`, `Parent.Type = single Type`, `Id = single ID`, or `Id IN (list of IDs)`. See Comparison Operators in the *Salesforce SOQL and SOSL Reference Guide* for a sample query.

A SOQL query must filter using one of the following Parent or Id clauses.

- `ParentId = [single ID]`
- `Parent.Type = [single type]`
- `Id = [single ID]`
- `Id IN = [list of IDs]`

SEE ALSO:

[Idea](#)
[IdeaComment](#)

WebLink

Represents a custom link to a URL or Scontrol.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`,
`update()`, `upsert()`

Special Access Rules

- To create a custom link, the client application must be logged in with the “Customize Application” permission.
- Customer Portal users can’t access this object.

Fields

Field Name	Details
Availability	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Description of the custom link. Limit is 1,000 characters.</p>
DisplayType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Type of display: button, link, or mass-action button.</p>
EncodingKey	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. Encoding of parameters on the URL link.</p>
HasMenubar	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the popup window shows a menu bar (<code>true</code>) or not (<code>false</code>).</p>
HasScrollbars	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the popup window shows scroll bars (<code>true</code>) or not (<code>false</code>).</p>
HasToolbar	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	<p>Indicates whether the popup window shows browser toolbars (<code>true</code>) or not (<code>false</code>). Toolbars normally contain navigation buttons like Back, Forward, and Print.</p>
Height	Type
	int
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Height of the popup in pixels.
IsProtected	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	<p>Indicates whether the object is protected (<code>true</code>) or not (<code>false</code>). Protected components that have been installed in other organizations can't be linked to or referenced by components created in the subscriber organization. A developer can easily delete a protected component contained in a managed package in a future release of the package without worrying about failing installations. However, once a component is marked as unprotected and is released globally, the developer can't delete it.</p>
IsResizable	Type
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates whether users are allowed to resize the popup window (<code>true</code>) or not (<code>false</code>).
LinkType	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	Required. Type of link (S-control or URL).
MasterLabel	Type
	string

Field Name	Details
	<p>Properties Create, Filter, Group, Nullable, Sort, Update</p>
	<p>Description Master label for the link. Limit is 240 characters. This display value is the internal label that is not translated.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. Name to display on page.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nullable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation.</p>
	<p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
	<p>This field can't be accessed unless the logged-in user has the "Customize Application" permission.</p>
OpenType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Required. How the custom link opens when clicked in a browser—NewWindow, Sidebar, or NoSidebar.</p>

Field Name	Details
PageOrObjectType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description Required. For standard objects, the name of the page on which to display the custom link. For custom objects, the name of the object.</p>
Position	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nullable, Restricted picklist, Sort, Update</p> <p>Description Location on the screen where the popup should open—TopLeft, FullScreen, or None.</p>
RequireRowSelection	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the custom link requires a row selection (<code>true</code>) or not (<code>false</code>).</p>
ScontrolId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description ID of the custom s-control object (Scontrol) to link to. Can include fields as tokens within the custom s-control object. Label is Custom S-Control ID.</p>
ShowsLocation	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the popup window shows the browser's address bar containing the URL (<code>true</code>) or not (<code>false</code>).</p>
ShowsStatus	<p>Type boolean</p>

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Show the status bar at the bottom of the browser.</p>
Url	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Required. URL of the page to link to. Can include fields as tokens within the URL. Limit: 1,024 KB.</p>
Width	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Width of the popup in pixels.</p>

Usage

Use this object to programmatically manage custom links, which allow client applications to integrate data with external URLs, an organization's intranet, or other back-end office systems. A custom link can point to:

- An external URL, such as `www.google.com` or your company's intranet.
- A custom s-control, such as a Java applet or Active-X control.

Custom links can include fields as tokens within the URL or custom s-control.

SEE ALSO:

[Scontrol](#)

WebLinkLocalization

Represents the translated value of the field label for a custom link to a URL or s-control when the Translation Workbench is enabled for your organization.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

- Your organization must be using Professional, Enterprise, Developer, Unlimited, or Performance Edition and be enabled for the Translation Workbench.
- To view this object, you must have the "View Setup and Configuration" permission.

Fields

Field	Details
LanguageLocaleKey	<p>Type picklist</p> <p>Properties Create, Filter, Nullable, Restricted picklist</p> <p>Description This field is available in API version 16.0 and earlier. It is the same as the Language field.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description This field is available in API version 17.0 and later. The combined language and locale ISO code, which controls the language for labels displayed in an application. This picklist contains the following fully-supported languages:</p> <ul style="list-style-type: none">• Chinese (Simplified): zh_CN• Chinese (Traditional): zh_TW• Danish: da• Dutch: nl_NL• English: en_US• Finnish: fi• French: fr• German: de• Italian: it• Japanese: ja• Korean: ko• Norwegian: no• Portuguese (Brazil): pt_BR• Russian: ru• Spanish: es

Field	Details
	<ul style="list-style-type: none">• Spanish (Mexico): es_MX• Swedish: sv• Thai: th
	<p>The following end-user only languages are available.</p> <ul style="list-style-type: none">• Arabic: ar• Bulgarian: bg• Croatian: hr• Czech: cs• English (UK): en_GB• Greek: el• Hebrew: iw• Hungarian: hu• Indonesian: in• Polish: pl• Portuguese (Portugal): pt_PT• Romanian: ro• Slovak: sk• Slovenian: sl• Turkish: tr• Ukrainian: uk• Vietnamese: vi
	<p>The following platform languages are available for organizations that use Salesforce exclusively as a platform.</p> <ul style="list-style-type: none">• Albanian: sq• Arabic (Algeria): ar_DZ• Arabic (Bahrain): ar_BH• Arabic (Egypt): ar_EG• Arabic (Iraq): ar_IQ• Arabic (Jordan): ar_JO• Arabic (Kuwait): ar_KW• Arabic (Lebanon): ar_LB• Arabic (Libya): ar LY• Arabic (Morocco): ar_MA• Arabic (Oman): ar_OM• Arabic (Qatar): ar_QA• Arabic (Saudi Arabia): ar_SA• Arabic (Sudan): ar_SD• Arabic (Syria): ar_SY

Field	Details
	<ul style="list-style-type: none">• Arabic (Tunisia): ar_TN• Arabic (United Arab Emirates): ar_AE• Arabic (Yemen): ar_YE• Armenian: hy• Basque: eu• Bosnian: bs• Bengali: bn• Chinese (Simplified—Singapore): zh_SG• Chinese (Traditional—Hong Kong): zh_HK• English (Australia): en_AU• English (Canada): en_CA• English (Hong Kong): en_HK• English (India): en_IN• English (Ireland): en_IE• English (Malaysia): en_MY• English (Philippines): en_PH• English (Singapore): en_SG• English (South Africa): en_ZA• Estonian: et• French (Belgium): fr_BE• French (Canada): fr_CA• French (Luxembourg): fr_LU• French (Switzerland): fr_CH• Georgian: ka• German (Austria): de_AT• German (Luxembourg): de_LU• German (Switzerland): de_CH• Hindi: hi• Icelandic: is• Irish: ga• Italian (Switzerland): it_CH• Latvian: lv• Lithuanian: lt• Luxembourgish: lb• Macedonian: mk• Malay: ms• Maltese: mt• Romanian (Moldova): ro_MD

Field	Details
	<ul style="list-style-type: none">• Montenegrin: sh_ME• Romansh: rm• Serbian (Cyrillic): sr• Serbian (Latin): sh• Spanish (Argentina): es_AR• Spanish (Bolivia): es_BO• Spanish (Chile): es_CL• Spanish (Colombia): es_CO• Spanish (Costa Rica): es_CR• Spanish (Dominican Republic): es_DO• Spanish (Ecuador): es_EC• Spanish (El Salvador): es_SV• Spanish (Guatemala): es_GT• Spanish (Honduras): es_HN• Spanish (Nicaragua): es_NI• Spanish (Panama): es_PA• Spanish (Paraguay): es_PY• Spanish (Peru): es_PE• Spanish (Puerto Rico): es_PR• Spanish (United States): es_US• Spanish (Uruguay): es_UY• Spanish (Venezuela): es_VE• Tagalog: tl• Tamil: ta• Urdu: ur• Welsh: cy

The values in this field are not related to the default locale selection.

NamespacePrefix	Type string
	Properties Filter, Group, Nillable, Sort
	Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix_componentName notation.
	The namespace prefix can have one of the following values:

Field	Details
	<ul style="list-style-type: none">• In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer.• In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Value	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The actual translated label of the custom link. Label is Translation.</p>
WebLinkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the WebLink that is being translated.</p>

Usage

Use this object to translate your custom links to URLs or s-controls into the different languages supported by Salesforce. Users with the Translation Workbench enabled can view custom link translations, but either the “Customize Application” or “Manage Translation” permission is required to create or update custom link translations.

SEE ALSO:

[CategoryNodeLocalization](#)
[ScontrolLocalization](#)

WorkAccess

Used to grant or restrict user access to give badge definitions. Each badge definition record must have one WorkAccess record.

Supported Calls

`create()`,`delete()`,`describeSObjects()`,`getDeleted()`,`getUpdated()`,`query()`,`retrieve()`,`undelete()`,`update()`,`upsert()`

Additional Considerations and Related Objects

WorkAccess is not available through Schema Builder and is not customizable. A WorkAccess record is **required** for users to **Give** BadgeDefinitions. If a WorkAccess record is not created, BadgeDefinitions will not be available to users.

The sharing of WorkAccess records is through [WorkAccessShare](#) on page 2015 For each WorkBadgeDefinition record, you must create both a WorkAccess record (per WorkBadgeDefinition) and WorkAccessShare records for sharing to users or groups.

Fields

Field Name	Details
AccessType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Define the type of Access given to user ("Give").</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Salesforce unique ID for owner of Access record.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Salesforce unique ID for BadgeDefinition record associated with this Access record.</p>

WorkAccessShare

Used to control Givers of WorkBadgeDefinition records.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Additional Considerations and Related Objects

Related to [WorkAccess Object](#). WorkAccess is the parent of WorkAccessShare.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description CRUD Access Level (picklist values: Read Only, Read/Write, Owner).</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID for WorkAccess record.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Owner or Manual sharing. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description User or Group ID for WorkAccess.</p>

WorkBadge

Represents information about who the badge was given to and which badge was given. A WorkBadge record is created for each recipient of a WorkBadgeDefinition.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Additional Considerations and Related Objects

WorkBadge is a lookup to WorkThanks. Each WorkBadge record must derive a Sourceld from WorkThanks. There can be multiple WorkBadge records tied to a single WorkThanks record.

Fields

Field Name	Details
DefinitionId	Type reference Properties Create, Filter, Group, Sort Description Required. Salesforce unique ID for the given WorkBadgeDefinition record given.
Description	Type textarea Properties Nillable Description The description of the WorkBadgeDefinition.
GiverId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the badge giver.
ImageUrl	Type url Properties Filter, Group, Nillable, Sort Description The URL of the badge image.
LastReferencedDate	Type dateTime

Field Name	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description The time stamp that indicates when the current user last viewed a record that is related to this WorkBadge.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The time stamp that indicates when the current user last viewed this WorkBadge. If this value is null, this record might have been only referenced (LastReferencedDate) and not viewed.</p>
Message	<p>Type textarea</p> <p>Properties Nillable</p> <p>Description The message accompanying the thanks badge.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the community that this WorkBadge is associated with. This field is available only if Salesforce Communities is enabled in your organization.</p>
RecipientId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Required. Salesforce unique ID for User who is the Recipient of Badge.</p>
RewardId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description Salesforce unique ID for Reward given with badge (if Reward Badge)</p>
SourceId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Salesforce unique ID for Thanks record referenced to this badge.</p>

WorkBadgeDefinition

Represents the attributes of a badge including the badge name, description, and image. Each WorkBadge record must have a lookup to a WorkBadgeDefinition since badge attributes (like badge name) are derived from the WorkBadgeDefinition object.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Additional Considerations and Related Objects

WorkBadgeDefinition has a field called `ImageUrl` that references a DocumentID. This is a required field for creating a Badge.

To grant "giver" access to a WorkBadgeDefinition, you must also create the [WorkAccess](#) (and the related [WorkAccessShare](#)) records.

Each WorkBadgeDefinition has an `ImageUrl` field that must be populated with a DocumentID of the Document record containing the badge image.

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description Required. Limit: 4000 characters. The description of the badge and what it means to receive this badge.</p>
GivenBadgeCount	<p>Type int</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of badges given per user or across all users.</p> <p> Note: This field can't be added in a list view or referenced in a formula field.</p>
ImageUrl	<p>Type url</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Required. This is the badge image that will be displayed in the UI. Use DocumentID or ImageURL.</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Represents whether a WorkBadgeDefinition is active and available in the UI.</p>
IsCompanyWide	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Represents a special class of badges known as Company Badges. Company badges are visible to the entire company and visible in specific list view filters.</p> <p> Note: If this field is selected, everyone within the user's network will be able to give the badge automatically. If this field is not selected, people with sharing must be added to the badge's access list in order to give the badge.</p>
IsLimitPerUser	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the badge limit is per user (<code>true</code>) or across all users (<code>false</code>). The default value is <code>false</code>.</p>

Field Name	Details
IsRewardBadge	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the badge is a reward badge (<code>true</code>) or not (<code>false</code>).</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The time stamp that indicates when the current user last viewed a record that is related to this WorkBadgeDefinition.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nullable, Sort</p> <p>Description The time stamp that indicates when the current user last viewed this WorkBadgeDefinition. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.</p>
LimitNumber	<p>Type int</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The badge limit per user or across all users.</p>
LimitStartDate	<p>Type dateTime</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The start date of the badge limit. The date can be reset to the current date.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p>

Field Name	Details
	<p>Description Required. Name of the Badge. Label: Badge Title.</p>
NetworkId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The ID of the community that this WorkBadgeDefinition is associated with. This field is available only if Salesforce Communities is enabled in your organization.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Salesforce User ID for User who is the Owner of the WorkBadgeDefinition record (usually the creator of the record)</p>
RewardFundId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Salesforce unique ID for the WorkRewardFund that is associated with this WorkBadgeDefinition. WorkBadgeDefinition records with a RewardFundId indicate a Reward Badge.</p>

WorkBadgeDefinitionHistory

Represents the history of changes to the values in the fields of a WorkBadgeDefinition object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	Type picklist Properties Filter, Group, Restricted picklist, Sort Description Type of change (e.g., created).
NewValue	Type Any Type Properties Nillable, Sort Description Updated value of record.
OldValue	Type Any Type Properties Nillable, Sort Description Previous value of record.
WorkBadgeDefinitionId	Type reference Properties Filter, Group, Sort Description ID of WorkBadgeDefinition record.

WorkBadgeDefinitionShare

Represents a sharing entry on a WorkBadgeDefinition object.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the work badge definition. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records <p>This field must be set to an access level that is higher than the organization's default access level for work badge definitions.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the WorkBadgeDefinition object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Valid values include:</p> <ul style="list-style-type: none"> • Owner: User is the owner of the WorkBadgeDefinition or is in a user role above the WorkBadgeDefinition owner in the role hierarchy. • Manual: User or group has access, because a user with "All" access manually shared the WorkBadgeDefinition with the user or group. • Rule: User or group has access via a WorkBadgeDefinition sharing rule.

Field Name	Details
UserOrGroupId	Type reference Properties Create, Filter, Group, Sort Description ID of the user or group that was given access to the work badge definition. This field can't be updated.

WorkCoaching

Represents a single coaching relationship between two users. One of the users is defined as the coach and the other is defined as a coachee. WorkCoaching is feed-enabled so there is a private feed available to the coach and coachee.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
CoachId	Type reference Properties Create, Filter, Group, Sort, Update Description [Required] The coach in this 1:1 coaching relationship.
CoachedId	Type reference Properties Create, Filter, Group, Sort, Update Description [Required] The user being coached in this 1:1 coaching relationship.
IsInactive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description
	Indicates whether the coaching relationship is <i>Inactive</i> (true) or not (false).
LastReferencedDate	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed a record that is related to this coaching relationship.
LastViewedDate	Type dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed this coaching relationship. If this value is null, this record might have been only referenced (LastReferencedDate) and not viewed.
Name	Type string
	Properties
	Create, Filter, Group, Sort, Update
	Description
	[Required] The record's name. Max length is 255 characters.
OwnerId	Type reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	ID of the contact who owns the WorkCoaching record.

WorkCoachingFeed

Represents a single feed item in the feed on the detail page for a coaching record.

A coaching feed shows recent changes to a lead record for any fields that are tracked in feeds, and posts and comments about the record. It's a useful way to stay up-to-date with changes made to coaching in Salesforce.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You can delete all feed items that you created. To delete feed items that you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the WorkCoachingFeed object
- “Moderate Chatter”

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file that is uploaded to the feed. Required if Type is ContentPost. The name of the file that is uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The size of the file (in bytes) that is uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The MIME type of the file that is uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, InsertedById is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The number of FeedLikes that are associated with this feed item.</p>
LinkUrl	<p>Type</p> <p>url</p> <p>Properties</p> <p>Nillable, Sort</p> <p>Description</p> <p>The URL of a LinkPost.</p>
NetworkScope	<p>Type</p> <p>picklist</p>

Field Name	Details
	<p>Properties Group, Nillable, Restricted picklist, Sort</p>
	<p>Description Specifies whether this feed item is available in the default community, a specific community, or all communities. If Salesforce Communities is enabled for your organization, this field is available in API version 28.0 and later.</p>
	<p><code>NetworkScope</code> can have the following values.</p>
	<ul style="list-style-type: none"> • <code>NetworkId</code>: The ID of the community in which the FeedItem is available. If left empty, the feed item is available only in the default community. • <code>AllNetworks</code>: The feed item is available in all communities.
	<p>Note the following exceptions for <code>NetworkScope</code>.</p>
	<ul style="list-style-type: none"> • Only feed items with a CollaborationGroup or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>.
<code>ParentId</code>	<p>Type reference</p>
	<p>Properties Filter, Group, Sort</p>
	<p>Description ID of the WorkCoaching record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
<code>RelatedRecordId</code>	<p>Type reference</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost. For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of type ContentPost.</p>
<code>Title</code>	<p>Type string</p>
	<p>Properties Group, Nillable, Sort</p>
	<p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field.

Field Name	Details
	<ul style="list-style-type: none">UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">AttachArticleEvent—generated event when a user attaches an article to a case.CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.CaseCommentPost—generated event when a user adds a case comment for a case object.ChangeStatusPost—generated event when a user changes the status of a case.ChatTranscriptPost—generated event when Live Agent transcript is saved to a case.EmailMessageEvent—generated event when an email related to a case object is sent or received.FacebookPost—generated when a Facebook post is created from a case. Deprecated.MilestoneEvent—generated when a case milestone is completed or reaches violation status.SocialPost—generated when a social post is created from a case.
Visibility	<p> Note: If you set Type to ContentPost, also specify ContentData and ContentFileName.</p> <p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The visibility of a FeedItem:<ul style="list-style-type: none">AllUsers: visible to all usersInternalUsers: visible to internal users</p>

WorkCoachingHistory

Represents the history of changes to the values in the fields of a WorkCoaching object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The name of the field that was changed.
NewValue	Type anyType Properties Nillable, Sort Description The new value of the field that was changed.
OldValue	Type anyType Properties Nillable, Sort Description The latest value of the field before it was changed.
WorkCoachingId	Type reference Properties Filter, Group, Sort Description The ID of the WorkCoaching object.

WorkCoachingShare

Represents a sharing entry on a WorkCoaching object.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the coaching relationship. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records. <p>This field must be set to an access level that is higher than the organization's default access level for coaching relationships.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the WorkCoaching object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none"> • Owner: User is the owner of the coaching relationship or is in a user role above the coaching relationship owner in the role hierarchy. • Manual: User or group has access, because a user with "All" access manually shared the coaching relationship with the user or group. • Rule: User or group has access via a sharing rule.

Field Name	Details
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the coaching relationship. This field can't be updated.</p>

WorkFeedback

Represents the answer to a question that a person was asked via a feedback request. Also used to store offered feedback without linking it to a particular question.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Additional Considerations and Related Objects

- Ownership is transferred to the requester on submit for certain types (ad-hoc feedback).
- The record is read-only after the request that it's linked to is set to Submitted.
- You can't link a feedback object to a request unless you are the recipient.
- The question that the feedback is linked to must be part of the same question set that the request is linked to.

Fields

Field Name	Details
Feedback	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Contains either the free-form text of the answer, or the choice selected by the user. Max length is 65536.</p>
Name	<p>Type string</p>

Field Name	Details
	<p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The name of the WorkFeedback record.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the WorkFeedback record.</p>
QuestionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort</p> <p>Description The question this answer applies to. When this feedback is linked to a request of an unsolicited type, the question ID is null.</p>
RequestId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the request this response belongs to, in case of offered feedback.</p>

WorkFeedbackHistory

Represents the history of changes to the values in the fields of a WorkFeedback object. Access is read-only.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
WorkFeedbackId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the WorkFeedback object.</p>

WorkFeedbackQuestion

Represents a free-form text type or multiple choice question within a set of questions.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
Choices	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description New-line separated list of valid choices for multiple choice questions. Maximum length is 1000 characters.</p>
Detail	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description Detailed instructions on how to answer the question.</p>
IsConfidentialAnswer	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Answers to questions marked confidential will not be shared with the subject of the review. This field applies only to performance summaries.</p>
IsOptional	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If this option is selected, the question is optional and isn't required to be answered. This field applies only to performance summaries.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description A short description of the question, which can be used as a header for reports and Calibration.</p>

Field Name	Details
Number	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The order of the question that is displayed within the question set, such as question number three in a question set that has five questions.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the WorkFeedbackQuestion.</p>
QuestionSetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The question set this question is a part of.</p>
Text	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The body of the question. Max length is 16384 characters.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Allows for either a free-form text answer or a multiple choice question defined by new-line separate choices in the 'Choices' field. Valid picklist values are:</p> <ul style="list-style-type: none"> • MultipleChoice • FreeText • Rating

WorkFeedbackQuestionHistory

Represents the history of changes to the values in the fields of a WorkFeedbackQuestion.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
WorkFeedbackQuestionId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the WorkFeedbackQuestion.</p>

WorkFeedbackQuestionSet

Represents a set of questions being asked. The question set is used to link all the individual requests where different recipients were asked the same set of questions on the same subject.

In the Work.com performance application, a question set defines the type of summaries and their due dates that will accompany the deployment of a specific performance summary cycle.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
DueDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The date that this specific question set is expected to be submitted by the recipient. This field applies only to performance summaries.</p>
FeedbackType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nullable, Restricted picklist, Sort</p> <p>Description The description of the collection of questions that are written in context to the type of recipient answering them; relative to the subject of the summary. This field applies only to performance summaries.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The name of the question set. Maximum length is 225 characters.</p>
OwnerId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the WorkFeedbackQuestionSet.</p>
PerformanceCycleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description If a question set is associated to a performance summary cycle, then that cycle ID is referenced in this field. This field applies only to performance summaries.</p>

WorkFeedbackQuestionSetHistory

Represents the history of changes to the values in the fields of a WorkFeedbackQuestionSet object. Access is read-only.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>

Field Name	Details
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
WorkFeedbackQuestionSetId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the WorkFeedbackQuestionSet object.</p>

WorkFeedbackQuestionSetShare

Represents a sharing entry on a WorkFeedbackQuestionSet.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the feedback question set. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records.

Field Name	Details
	This field must be set to an access level that is higher than the organization's default access level for feedback question sets.
ParentId	Type reference Properties Create, Filter, Group, Sort Description ID of the WorkFeedbackQuestionSet object that is associated with this sharing entry.
RowCause	Type picklist Properties Filter, Group, Restricted picklist, Sort Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include: <ul style="list-style-type: none">• Owner: User is the owner of the WorkFeedbackQuestionSet or is in a user role above the WorkFeedbackQuestionSet owner in the role hierarchy.• Manual: User or group has access, because a user with "All" access manually shared the WorkFeedbackQuestionSet with the user or group.• Rule: User or group has access via a WorkFeedbackQuestionSet sharing rule.
UserOrGroupId	Type reference Properties Create, Filter, Group, Sort Description ID of the user or group that was given access to the feedback question set. This field can't be updated.

WorkFeedbackQuestionShare

Represents a sharing entry on a WorkFeedbackQuestion.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the feedback question. The possible values are:<ul style="list-style-type: none">• Read• Edit• All: This value is not valid when you create, update, or delete records.This field must be set to an access level that is higher than the organization's default access level for feedback questions.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the WorkFeedbackQuestion object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:<ul style="list-style-type: none">• Owner: User is the owner of the WorkFeedbackQuestion or is in a user role above the WorkFeedbackQuestion owner in the role hierarchy.• Manual: User or group has access, because a user with "All" access manually shared the WorkFeedbackQuestion with the user or group.</p>

Field Name	Details
	<ul style="list-style-type: none"> Rule: User or group has access via a WorkFeedbackQuestion sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the feedback question. This field can't be updated.</p>

WorkFeedbackRequest

Represents a single feedback request on a subject or topic (question) to a single recipient in the feedback application. In the case of offered feedback, WorkFeedbackRequest represents feedback that is offered about a subject. In the performance application, WorkFeedbackRequest represents a request for feedback on a set of questions from a question set, on a subject—for the recipient to complete and submit.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Additional Considerations and Related Objects

- After a request's state is changed to Submitted, fields can't be changed, except for LastSharedDate and IsUnreadByOwner.
- If LastRemindDate is updated, a reminder notification will be sent to the request's recipient (only possible when request is in Draft state).
- When a new request is created, a notification is sent to the recipient.
- When a recipient of a request submits their feedback (Draft->Submitted), a notification will be sent to requester (except for offered feedback).
- Requester cannot modify the subject of the question set after a request is created.
- For offered feedback (to user, to manager, or both), the person who is offering feedback is both the creator of WorkFeedbackRequest as well as the recipient.

Fields

Field Name	Details
AdHocFeedback	<p>Type textarea</p>

Field Name	Details
	Properties
	Create, Filter, Nillable, Sort
	Description
	The content of the feedback.
AdHocQuestion	Type
	textarea
	Properties
	Create, Filter, Nillable, Sort
	Description
	The content of the feedback question.
Description	Type
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The description of the WorkFeedbackRequest.
FeedbackRequestState	Type
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update
	Description
	The current state of the feedback request. Allowed picklist values are:
	<ul style="list-style-type: none"> • Draft • Submitted • Declined
FeedbackType	Type
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Restricted picklist, Sort
	Description
	Specifies the type of request. Picklist values that are used for performance summaries:
	<ul style="list-style-type: none"> • Unspecified • Peer Summary • Self Summary • Manager Summary • Skip Level Summary

Field Name	Details
	<p>Picklist values that are used for feedback:</p> <ul style="list-style-type: none">● Personal● Unsolicited to User● Unsolicited to Manager● Unsolicited to User and Manager● On Topic <p>The type of the feedback determines the sharing and visibility rules that are applied to answers.</p>
IsDeployed	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If true, the feedback is part of a deployed performance summary cycle.</p>
IsShareWithSubject	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If true, the feedback is shared with the summary subject.</p>
IsUnreadByOwner	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If true, the submitted request has not been seen by the requester.</p>
IsUnsolicited	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description If true, the feedback request is unsolicited feedback offered to another user.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	<p>Description The time stamp that indicates when the current user last viewed a record that is related to this WorkFeedbackRequest.</p>
LastRemindDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The last time a reminder was sent to the recipient of this draft request.</p>
LastSharedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The last time this request was shared with another user or group.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The time stamp that indicates when the current user last viewed this WorkFeedbackRequest. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The name of the WorkFeedbackRequest.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the owner of the WorkFeedbackRequest.</p>

Field Name	Details
PerformanceCycleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Used by performance summaries to link to a summary cycle. This field applies only to performance summaries.</p>
QuestionSetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Question set associated with the current request.</p>
RecipientId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description User asked to provide feedback on the subject.</p>
RelatedObjectId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description Specifies a record in the system that this feedback request is related to. Used by ad-hoc feedback to gather feedback in the context of an opportunity or Work.com goal. Used by performance summaries to link to a summary cycle.</p>
SharingScope	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The users that see the feedback. SharingScope can have the following values:</p> <ul style="list-style-type: none"> • Nobody • Subject

Field Name	Details
	<ul style="list-style-type: none"> Manager SubjectAndManager
SubjectId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the user that this request (or offer) is about.</p>
SubmitFeedbackToId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the person this performance summary feedback request (and its respective answers) is shared with. It's also the ID of the person who owns the requested subject's manager summary request. This field applies only to performance summaries.</p>
SubmittedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The last time (in case it was reopened by admin) this request was submitted by the recipient. This field applies only to performance summaries.</p>

WorkFeedbackRequestFeed

Represents a single feed item in the feed displayed on the feedback request detail page.

The feedback request feed shows changes to a request for fields that are tracked in feeds, posts, and comments about the request.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The file uploaded to the feed. Required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>

Field Name	Details
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates posts and comments from another application into a feed, then InsertedById is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item Body contains rich text. Set IsRichText to true if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u>

Field Name	Details
	<ul style="list-style-type: none"> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the goal record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p>

Field Name	Details
	<p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API</p> <ul style="list-style-type: none">ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.AdvancedTextPost—created when a user posts a group announcement.AnnouncementPost—Not used.ApprovalPost—generated when a user submits an approval.BasicTemplateFeedItem—Not used.CanvasPost—a post made by a canvas app posted on a feed.CollaborationGroupCreated—generated when a user creates a public group.CollaborationGroupUnarchived—Not used.ContentPost—a post with an attached file.CreatedRecordEvent—generated when a user creates a record from the publisher.DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold.DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed.

Field Name	Details
	<ul style="list-style-type: none">• LinkPost—a post with an attached URL.• PollPost—a poll posted on a feed.• ProfileSkillPost—generated when a skill is added to a user's Chatter profile.• QuestionPost—generated when a user posts a question.• ReplyPost—generated when Chatter Answers posts a reply.• RipplePost—generated when a user creates a Thanks badge in Work.com.• TextPost—a direct text entry on a feed.• TrackedChange—a change or group of changes to a tracked field.• UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the <code>Type</code> picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none">• AttachArticleEvent—generated event when a user attaches an article to a case.• CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.• CaseCommentPost—generated event when a user adds a case comment for a case object.• ChangeStatusPost—generated event when a user changes the status of a case.• ChatTranscriptPost—generated event when Live Agent transcript is saved to a case.• EmailMessageEvent—generated event when an email related to a case object is sent or received.• FacebookPost—generated when a Facebook post is created from a case. Deprecated.• MilestoneEvent—generated when a case milestone is completed or reaches violation status.• SocialPost—generated when a social post is created from a case.

WorkFeedbackRequestHistory

Represents the history of changes to the values in the fields of a `WorkFeedbackRequest`.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The name of the field that was changed.
NewValue	Type anyType Properties Nillable, Sort Description The new value of the field that was changed.
OldValue	Type anyType Properties Nillable, Sort Description The latest value of the field before it was changed.
WorkFeedbackRequestId	Type picklist Properties Filter, Group, Sort Description ID of the WorkFeedbackRequest.

WorkFeedbackRequestShare

Represents a sharing entry on a WorkFeedbackRequest.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the feedback request. The possible values are:<ul style="list-style-type: none">• Read• Edit• All: This value is not valid when you create, update, or delete records.This field must be set to an access level that is higher than the organization's default access level for feedback requests.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the WorkFeedbackRequest object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:<ul style="list-style-type: none">• Owner: User is the owner of the WorkFeedbackRequest or is in a user role above the WorkFeedbackRequest owner in the role hierarchy.• Manual: User or group has access, because a user with "All" access manually shared the WorkFeedbackRequest with the user or group.• Rule: User or group has access via a WorkFeedbackRequest sharing rule.</p>

Field Name	Details
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the feedback request. This field can't be updated.</p>

WorkFeedbackShare

Represents a sharing entry on a WorkFeedback object.

Supported Calls

`create(), delete(), describeObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the feedback. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records. <p>This field must be set to an access level that is higher than the organization's default access level for feedback.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>ID of the WorkFeedback object that is associated with this sharing entry.</p>
RowCause	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Restricted picklist, Sort</p> <p>Description</p> <p>Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p> <p>Valid values include:</p> <ul style="list-style-type: none">• Owner: User is the owner of the WorkFeedback or is in a user role above the WorkFeedback owner in the role hierarchy.• Manual: User or group has access, because a user with "All" access manually shared the WorkFeedback with the user or group.• Rule: User or group has access via a WorkFeedback sharing rule.
UserOrGroupId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Sort</p> <p>Description</p> <p>ID of the user or group that was given access to the feedback. This field can't be updated.</p>

WorkGoal

Represents the components of a goal, such as its description and associated metrics. This object has been deprecated as of API version 35.0. Use the [Goal](#) object to query information about Work.com goals.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Related

[WorkGoalCollaborator](#), [WorkGoalLink](#), [WorkGoalFeed](#)

Fields

Field Name	Details
ActualValue	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The actual value of the WorkGoal metric. Applicable only to WorkGoal objects of Type: Metric.</p>
ActualValueExternalUrl	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Contains a URL that references work.com data synchronization for the actual value of a metric. Applicable only to WorkGoal objects of Type: Metric.</p>
CompletionDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The completion date of the goal.</p> <p> Note: Field-level security limits access to only administrators and owners by default, and only they can complete a goal.</p>
Description	<p>Type textarea (max length 4000)</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the goal.</p>
DueDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The date the WorkGoal object is due (optional). Applicable only to WorkGoal objects of Type: Metric.</p>

Field Name	Details
FlaggedAs	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The progress of the WorkGoal object. Applicable only to WorkGoal objects of Type: Metric. Possible values:</p> <ul style="list-style-type: none"> • On Track: Progress on the metric is on track. • Behind: Progress on the metric is behind schedule. • Postponed: The metric is postponed. • Critical: Progress on the metric is critical.
ImageUrl	<p>Type url</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The URL for the goal image. The image must be stored in Documents and set as externally available. Applicable only to WorkGoal objects of Type: Goal.</p>
InitialValue	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The initial value of the WorkGoal metric. Applicable only to WorkGoal objects of Type: Metric and MetricType: Progress or Percent.</p>
IsKeyCompanyGoal	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Used to indicate if the goal is a key company goal. Used for the Company Goal Showcase. Applicable only to WorkGoal objects of Type: Goal.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	Description
	<p>The time stamp that indicates when the current user last viewed a record that is related to this goal.</p>
LastSyncDate	Type
	dateTime
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	<p>The time stamp that indicates when the actual value was last synced with the associated metrics report.</p>
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	<p>The time stamp that indicates when the current user last viewed this goal.</p>
MetricType	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	<p>The type of metric that is represented. (See values in the following list). Applies only to WorkGoal objects of Type: Metric.</p>
	Possible values:
	<ul style="list-style-type: none"> • Progress: ActualValue / TargetValue as a percentage • Percent: the metric as a percentage only • YesNo: the completed / not completed metric as a milestone • Absolute: Deprecated
MetricTypeDataSource	Type
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	<p>Specifies how the metric (ActualValue and CurrentValue) is updated. Applies only to WorkGoal objects of Type: Goal and Metric.</p>
	Possible values:

Field Name	Details
	<ul style="list-style-type: none"> Manual: indicates that the actual and target value of the metric is updated manually by the user Rollup: indicates that the actual and target value of a goal is rolled up automatically by Work.com Goals DataSyncActualOnly: indicates that the actual value of the metric is linked to a Salesforce report
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The name of the WorkGoal object. (Maximum length is 255.)</p>
OverallStatus	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The overall calculated status of the WorkGoal based on FlaggedAs and CompletionDate.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description ID of the user who owns the WorkGoal.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Specifies the structural parent of the WorkGoal. For example, a goal that has a metric is represented by a WorkGoal of Type Metric, which has a parent of WorkGoal of Type Goal.</p> <p> Note: The root and the parent must be set to the parent goal for any child metrics.</p>
Progress	<p>Type percent</p>

Field Name	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Read Only. The overall progress of the WorkGoal.</p>
RootId	<p>Type reference to a WorkGoal object</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Specifies the structural root of the WorkGoal. For example, a goal that has a metric is represented by a WorkGoal of Type Metric, which has a root of WorkGoal of Type Goal.</p>
State	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The state of the WorkGoal object. Applies only to WorkGoal objects of Type: Metric.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • Draft: the draft state for the WorkGoal • Published: published state for the WorkGoal • Archived: archived state for the WorkGoal (for example, goals that no longer apply)
TargetValue	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The target value of the WorkGoal. Applies only to WorkGoal objects of Type: Metric.</p>
Type	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>The type of the WorkGoal object, used to differentiate between the components of a goal. (This field is used to represent components of a goal such as its description and associated metrics.)</p> <p>Possible values:</p> <ul style="list-style-type: none">• Goal: a goal• Metric: a metric (typically associated with goals)• Objective: an objective• KeyResult: a key result (typically associated with objectives)• V2Mom: a V2MOM (pilot feature)• Vision: a vision (pilot feature — typically associated with V2MOM)• Value: a value (pilot feature - typically associated with V2MOM)• Method: a method (pilot feature - typically associated with V2MOM)• Obstacle: an obstacle (pilot feature - typically associated with V2MOM)• Measure: a measure (pilot feature - typically associated with a method) <p> Note: Administrators can rename goals and metrics to objectives and key results, respectively. If this preference is enabled, use the <code>Type</code> Objective or KeyResult. Otherwise, use the default <code>Type</code> Goal or KeyResult.</p>
Weight	<p>Type</p> <p>double</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>The weight of the goal or metric. The sum of the weights should equal 100%.</p>

WorkGoalCollaborator

Represents collaborators on a WorkGoal object. This doesn't include WorkGoal followers, which is handled by Chatter Feed Follow functionality. This object has been deprecated as of API version 35.0. Use the [Goal](#) object to query information about Work.com goals.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`,
`retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
InvitationDate	<p>Type date</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The date that a user was invited to become a collaborator (nill if the user was not invited).</p>
State	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Indicates the state of the collaborating user. Whether the user has not responded, joined, or declined collaboration. The possible values are:</p> <ul style="list-style-type: none"> • PendingResponse: a user who was invited to collaborate but hasn't joined or declined • Joined: a user who is collaborating on a goal (joined/commit) • Declined: a user who declined to collaborate on a goal
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The collaborating user.</p>
WorkGoalId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The WorkGoal object that this collaborator is a part of.</p>

WorkGoalCollaboratorHistory

Represents the history of changes to the values in the fields in a WorkGoalCollaborator object. Access is read-only.



Note: This object has been deprecated as of API version 35.0. Use the [Goal](#) object to query information about Work.com goals in API version 35.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Name of the standard or custom field.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description New value of the modified field.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description Previous value of the modified field.</p>
WorkGoalCollaboratorId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the WorkGoalCollaborator object that is associated with this history entry.</p>

WorkGoalFeed

Represents a single feed item in the feed displayed on the goal page for a Goal record. The goal feed shows changes to a goal for fields that are tracked in feeds, posts, and comments about the goal, and updates on metrics. This object has been deprecated as of API version 35.0. Use the [GoalFeed](#) object to query information about feed items for Work.com goals.

Supported Calls

`delete(), describesObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p>

Field Name	Details
	<p>Description Available in API version 36.0 and earlier only. The description of the file specified in ContentData.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The file uploaded to the feed. Required if Type is ContentPost. The name of the file uploaded to the feed. Setting ContentFileName automatically sets Type to ContentPost.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates multiple posts and comments from another application into a feed, then InsertedById is set to the ID of the logged-in user.</p>

Field Name	Details
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item Body contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p> <p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <code><p></code> <p> Tip: Though the <code>
</code> tag isn't supported, you can use <code><p>&nbsp;</p></code> to create lines.</p> <ul style="list-style-type: none"> • <code></code> • <code><code></code> • <code><i></code> • <code><u></code> • <code><s></code> • <code></code> • <code></code> • <code></code> • <code></code> <p>The <code></code> tag is accessible only via the API and must reference files in Salesforce similar to this example: <code></code></p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p>

Field Name	Details
NetworkScope	<p>Description</p> <p>The URL of a LinkPost.</p>
ParentId	<p>Type</p> <p>reference</p>
RelatedRecordId	<p>Type</p> <p>reference</p>

Properties	Group, Nillable, Restricted picklist, Sort
Description	<p>Specifies whether this feed item is available in the default community, a specific community, or all communities. If Salesforce Communities is enabled for your organization, this field is available in API version 28.0 and later.</p> <p>NetworkScope can have the following values:</p> <ul style="list-style-type: none"> • NetworkId—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • AllNetworks—The feed item is available in all communities. <p>Note the following exceptions for NetworkScope:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a NetworkId or a null value for NetworkScope. • For feed items with a record parent, users can set NetworkScope only to AllNetworks. • You can't filter a FeedItem on the NetworkScope field.

Properties	Filter, Group, Sort
Description	<p>ID of the goal record that is tracked in the feed. The feed is displayed on the detail page for this record.</p>

Properties	Group, Nillable, Sort
Description	<p>ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of TypeContentPost.</p>

Field Name	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the FeedItem. When the Type is LinkPost, the LinkUrl is the URL and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed.

Field Name	Details
	<ul style="list-style-type: none"> • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field. • UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> • AttachArticleEvent—generated event when a user attaches an article to a case. • CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • CaseCommentPost—generated event when a user adds a case comment for a case object. • ChangeStatusPost—generated event when a user changes the status of a case. • ChatTranscriptPost—generated event when Live Agent transcript is saved to a case. • EmailMessageEvent—generated event when an email related to a case object is sent or received. • FacebookPost—generated when a Facebook post is created from a case. Deprecated. • MilestoneEvent—generated when a case milestone is completed or reaches violation status. • SocialPost—generated when a social post is created from a case.
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The visibility of a FeedItem:</p> <ul style="list-style-type: none"> • AllUsers: visible to all users • InternalUsers: visible to internal users

WorkGoalHistory

Represents the history of changes to the values in the fields of a WorkGoal. Access is read-only. This object has been deprecated as of API version 35.0. Use the [GoalHistory](#) object to query historical information for Work.com goals.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type Any Type</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type Any Type</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
WorkGoalId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Goal. Label is Goal ID.</p>

WorkGoalLink

Represents the relationship between two goals (many to many relationship). This object has been deprecated as of API version 35.0. Use the [GoalLink](#) object to query information about the relationship between two Work.com goals.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), undelete(), update(), upsert()`

Fields

Field Name	Details
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Whether the WorkGoalLink is active (<code>true</code>) or not (<code>false</code>)</p>
LinkType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of link</p>
Name	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The auto-generated name of the goal link</p>
SourceGoalId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the source WorkGoal object</p>

Field Name	Details
TargetGoalId	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of the target WorkGoal object

WorkGoalShare

Represents a sharing entry on a WorkGoal object. This object has been deprecated as of API version 35.0. Use the [GoalShare](#) object to query information about sharing for Work.com goals.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The user's or group's level of access to the goal. The possible values are: <ul style="list-style-type: none">• Read• Edit• All: This value is not valid when you create, update, or delete records This field must be set to an access level that is higher than the organization's default access level for goals.
ParentId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	<p>Description ID of the WorkGoal object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none">• Owner: User is the owner of the WorkGoal or is in a user role above the WorkGoal owner in the role hierarchy.• Manual: User or group has access, because a user with "All" access manually shared the WorkGoal with the user or group.• Rule: User or group has access via a WorkGoal sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the goal. This field can't be updated.</p>

WorkOrder

Represents a task or series of tasks to be performed, typically in field service. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
AccountId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The account associated with the work order.</p>
Address	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address where the work order is completed.</p>
AssetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The asset associated with the work order.</p>
BusinessHoursId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The business hours associated with the work order.</p>
CaseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The case associated with the work order.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description The city where the work order is completed. Maximum length is 40 characters.</p>
ContactId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The contact associated with the work order.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country where the work order is completed. Maximum length is 80 characters.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is Currency ISO Code.</p>
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the work order. Try to include the steps needed to change the work order's status to Completed.</p>
Discount	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The weighted average of the discounts on all line items in the work order. It can be any positive number up to 100.</p>

Field Name	Details
Duration	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The estimated time required to complete the work order. Specify the duration unit in the Duration Type field.</p>
DurationType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The unit of the duration: Minutes or Hours.</p>
EndDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date when the work order is completed. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the EndDate to 365 days after the StartDate.</p>
EntitlementId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The entitlement associated with the work order.</p>
GeocodeAccuracy	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Accuracy level of the geocode for the address. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p> <p> Note: This field is available in the API only.</p>

Field Name	Details
GrandTotal	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The total price of the work order with tax added.</p>
IsClosed	<p>Type boolean</p> <p>Properties Group, Defaulted on create, Filter, Sort</p> <p>Description Indicates whether the work order is closed (<code>true</code>) or open (<code>false</code>).</p> <p> Tip: Use this field to report on closed versus open work orders.</p>
IsStopped	<p>Type boolean</p> <p>Properties Group, Defaulted on create, Create, Filter, Sort, Update</p> <p>Description Indicates whether a milestone is paused (<code>true</code>) or counting down (<code>false</code>). This field is available only if Enable stopped time and actual elapsed time is selected on the Entitlement Settings page.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the work order was last modified. Its label in the user interface is Last Modified Date.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the work order was last viewed.</p>
Latitude	<p>Type double</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with Longitude to specify the precise geolocation of the address where the work order is completed. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
	<p> Note: This field is available in the API only.</p>
LineItemCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of work order line items in the work order. Its label in the user interface is <code>Line Items</code>.</p>
Longitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with Latitude to specify the precise geolocation of the address where the work order is completed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.</p>
	<p> Note: This field is available in the API only.</p>
MilestoneStatus	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Indicates the status of a milestone. This field is visible if an entitlement process is applied to a work order.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description The work order's assigned owner.</p>
ParentWorkOrderId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The work order's parent work order, if it has one.</p> <p> Tip: Create a custom report to view a work order's child work orders.</p>
PostalCode	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The postal code where the work order is completed. Maximum length is 20 characters.</p>
Pricebook2Id	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The price book associated with the work order. Adding a price book to the work order lets you assign different price book entries to the work order's line items. This is only available if Product2 is enabled.</p>
Priority	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description The priority of the work order. The picklist includes the following values, which can be customized:</p> <ul style="list-style-type: none">• Low• Medium• High• Critical

Field Name	Details
RootWorkOrderId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The top-level work order in a work order hierarchy. Depending on where a work order lies in the hierarchy, its root could be the same as its parent.</p> <p> Note: View a work order's child work order in the Child Work Orders related list.</p>
ServiceContractId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The service contract associated with the work order.</p>
ServiceTerritoryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The service territory where the work order is taking place.</p>
SlaExitDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The time that the work order exits the entitlement process.</p>
SlaStartDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Update, Sort</p> <p>Description The time that the work order enters the entitlement process. You can update or reset the time if you have "Edit" permission on work orders.</p>
StartDate	<p>Type dateTime</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date when the work order goes into effect. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the StartDate to the date when the Status changes to In Progress.</p>
State	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state where the work order is completed. Maximum length is 80 characters.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description The status of the work order. The picklist includes the following values, which can be customized:</p> <ul style="list-style-type: none">• <i>New</i>—Work order was created, but there hasn't yet been any activity.• <i>In Progress</i>—Work has begun.• <i>On Hold</i>—Work is paused.• <i>Completed</i>—Work is complete.• <i>Cannot Complete</i>—Work could not be completed.• <i>Closed</i>—All work and associated activity is complete.• <i>Canceled</i>—Work is canceled, typically before any work began. <p>Changing a work order's status does not affect the status of its work order line items or associated service appointments.</p>
StatusCategory	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The category that each <code>Status</code> value falls into. The <code>Status Category</code> field has eight default values: seven values which are identical to the default <code>Status</code> values, and a <code>None</code> value for statuses without a status category.</p>

Field Name	Details
	<p>If you create custom Status values, you must indicate which category it belongs to. For example, if you create a <i>Waiting for Response</i> value, you may decide that it belongs in the <i>On Hold</i> category.</p> <p>The Status Category field can be useful to reference in custom apps, triggers, and validation rules. Status categories let you extend and customize the work life cycle while still maintaining a consistent work classification for tracking, reporting, and business process management.</p>
StopStartDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the milestone was paused. The label in the user interface is Stopped Since.</p>
Street	<p>Type textArea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street number and name where the work order is completed.</p>
Subject	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The subject of the work order. Try to describe the nature and purpose of the job to be completed. For example, "Annual On-Site Well Maintenance." Maximum length is 255 characters.</p>
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The total of the work order line items' subtotals before discounts and taxes are applied.</p>
Tax	<p>Type currency</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The total tax on the work order. You can enter a number with or without the currency symbol and use up to two decimal places. For example, in a work order whose total price is \$100, enter \$10 to apply a 10% tax.</p>
TotalPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The total of the work order line items' prices. This value has discounts applied but not tax.</p>
WorkOrderNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description An eight-digit, auto-generated number that identifies the work order.</p>
WorkTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The work type associated with the work order. When a work type is selected, the work order automatically inherits the work type's Duration, Duration Type, and required skills.</p>

Usage

A work order can be a child record of accounts, assets, cases, contacts, entitlements, service contracts, and other work orders. We recommend adding work order line items to the work order to break the main task into smaller subtasks.

WorkOrderFeed

Represents a single feed item on a work order record detail page. This object is available in API version 36.0 and later.

A work order feed shows changes to tracked fields on a work order record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to work orders in Salesforce.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Work orders or Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Work Order object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization. When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator.</p>

Field Name	Details
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The name of the file to upload to the feed. Required if <code>Type</code> is <code>ContentPost</code>. Setting this field sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. The size of the file uploaded to the feed in bytes. Determined during insert.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. The MIME type of the file uploaded to the feed. Determined during insert.</p>

Field Name	Details
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the work order record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p>

Field Name	Details
	<p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

WorkOrderHistory

Represents the history of changes made to tracked fields on a work order. This object is available in API version 36.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Work orders or Field Service Lightning must be enabled in your organization, and field tracking for work order fields must be configured.

Fields

Field Name	Details
Field	<p>Type picklist</p>

Field Name	Details
	<p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
WorkOrderId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the work order being tracked. The history is displayed on the detail page for this record.</p>

WorkOrderLineItem

Represents a subtask on a work order, typically in field service. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
Address	<p>Type address</p> <p>Properties Filter, Nillable</p> <p>Description The compound form of the address where the line item is completed.</p>
AssetId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The asset associated with the work order line item. The asset is not automatically inherited from the parent work order.</p>
City	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The city where the line item is completed. Maximum length is 40 characters.</p>
Country	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The country where the line item is completed. Maximum length is 80 characters.</p>
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is Currency ISO Code.</p>
Description	<p>Type textarea</p>

Field Name	Details
	<p>Properties Create, Nillable, Update</p> <p>Description The description of the work order line item. Try to describe the steps needed to mark the line item Completed.</p>
Discount	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The percent discount to apply to the line item. You can enter a number with or without the percent symbol, and you can use up to two decimal places.</p>
Duration	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The estimated time required to complete the line item. Specify the duration unit in the Duration Type field.</p>
DurationType	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The unit of the duration: Minutes or Hours.</p>
EndDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date on which the line item is completed. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the EndDate to 365 days after the StartDate.</p>
GeocodeAccuracy	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.</p> <p> Note: This field is available in the API only.</p>
IsClosed	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the line item has been closed. Changing the line item's status to Closed causes this checkbox to be selected in the user interface (sets IsClosed to true).</p> <p> Tip: Use this field to report on closed versus open work order line items.</p>
Latitude	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Used with Longitude to specify the precise geolocation of the address where the line item is completed. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>
LineItemNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description An auto-generated number that identifies the work order line item. Each work order's line items start at 1.</p>
ListPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	<p>Description</p> <p>The price of the line item (product) as listed in its corresponding price book entry. If a price book entry isn't specified, the list price defaults to zero.</p>
Longitude	<p>Type</p> <p>double</p> <p>Properties</p> <p>Create, Filter, Nillable, Sort, Update</p> <p>Description</p> <p>Used with <code>Latitude</code> to specify the precise geolocation of the address where the line item is completed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.</p> <p> Note: This field is available in the API only.</p>
OrderId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>The order associated with the line item. For example, you may need to order replacement parts before you can complete the line item.</p>
ParentWorkOrderLineItemId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>The line item's parent work order line item, if it has one.</p> <p> Tip: Create a custom report to view a line item's child line items.</p>
PostalCode	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>The postal code where the line item is completed. Maximum length is 20 characters.</p>
PricebookEntryId	<p>Type</p> <p>reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The price book entry (product) associated with the line item. The label in the user interface is <code>Product</code>. This field's lookup search only returns products that are included in the work order's price book.</p>
Product2Id	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The product associated with the price book entry. This field is not available in the user interface. For best results, use the <code>PricebookEntryId</code> field in any custom code or layouts.</p>
Quantity	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Number of units of the line item included in the associated work order.</p>
RootWorkOrderLineItemId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The top-level line item in a work order line item hierarchy. Depending on where a line item lies in the hierarchy, its root could be the same as its parent.</p> <p> Note: View a line item's child line items in the Child Work Order Line Items related list.</p>
ServiceTerritoryId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The service territory where the line item is completed.</p>
StartDate	<p>Type dateTime</p>

Field Name	Details
	<p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date on which the line item goes into effect. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the StartDate to the date when the Status changes to In Progress.</p>
State	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The state where the line item is completed. Maximum length is 80 characters.</p>
Status	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update</p> <p>Description The status of the line item. The picklist includes the following values, which can be customized:</p> <ul style="list-style-type: none"> • New—Line item was created, but there hasn't yet been any activity. • In Progress—Work has begun. • On Hold—Work is paused. • Completed—Work is complete. • Cannot Complete—Work could not be completed. • Closed—All work and associated activity is complete. • Canceled—Work is canceled, typically before any work began.
StatusCategory	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The category that each Status value falls into. The Status Category field has eight default values: seven values which are identical to the default Status values, and a None value for statuses without a status category. If you create custom Status values, you must indicate which category it belongs to. For example, if you create a Waiting for Response value, you may decide that it belongs in the On Hold category.</p>

Field Name	Details
	<p>The <code>Status Category</code> field can be useful to reference in custom apps, triggers, and validation rules. Status categories let you extend and customize the work life cycle while still maintaining a consistent work classification for tracking, reporting, and business process management.</p>
Street	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The street number and name where the line item is completed.</p>
Subject	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A word or phrase describing the line item.</p>
Subtotal	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description (Read only) The line item's unit price multiplied by the quantity.</p>
TotalPrice	<p>Type currency</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Read only. The line item's subtotal with discounts applied.</p>
UnitPrice	<p>Type currency</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Initially, the unit price for a work order line item is the line item's list price from the price book, but you can change it.</p>

Field Name	Details
WorkOrderId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The line item's parent work order. Because work order line items must be associated with a work order, this is a required field.</p>
WorkTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The work type associated with the line item. When a work type is selected, the line item automatically inherits the work type's Duration, Duration Type, and required skills.</p>

Usage

A work order line item is a child record of a work order. It represents a specific subtask on a work order.

For example, suppose a customer purchased a truck from you. The truck is represented as an asset in your Salesforce org. After some time, the truck needs both headlight bulbs replaced. Here's one way that you can use work orders and work order line items to track the repair.

1. Create a work order named "Replace Headlight Bulbs" from the asset record detail page.
2. Add three work order line items to the work order: "Replace Left Headlight Bulb," "Replace Right Headlight Bulb," and "Test Headlights."
3. Assign the work order to a technician via a queue.
4. As the technician completes each line item, he or she marks the item as Completed.
5. When all the line items are complete, the technician marks the work order as Completed.

WorkOrderLineItemFeed

Represents a single feed item on a work order line item record detail page. This object is available in API version 36.0 and later.

A work order line item feed shows changes to tracked fields on a work order line item record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to work order line items in Salesforce.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Work orders or Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- “Modify All Data”
- “Modify All” on the Work Order Line Item object
- “Moderate Chatter”



Note: Users with the “Moderate Chatter” permission can delete only the feed items and comments they see.

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
ConnectionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization. When a PartnerNetworkConnection modifies a record that is tracked, the <code>CreatedBy</code> field contains the ID of the system administrator.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p>

Field Name	Details
	Description
	Available in API version 36.0 and earlier only. Required if Type is ContentPost. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets Type to ContentPost.
ContentDescription	Type
	textarea
	Properties
	Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. The description of the file specified in ContentData.
ContentFileName	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. The name of the file to upload to the feed. Required if Type is ContentPost. Setting this field sets Type to ContentPost.
ContentSize	Type
	int
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. Read only. The size of the file uploaded to the feed in bytes. Determined during insert.
ContentType	Type
	string
	Properties
	Group, Nillable, Sort
	Description
	Available in API version 36.0 and earlier only. Read only. The MIME type of the file uploaded to the feed. Determined during insert.
InsertedById	Type
	reference
	Properties
	Group, Nillable, Sort

Field Name	Details
	<p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the work order line item record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>

Field Name	Details
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When Type = LinkPost, Link URL= the link URL and Title = the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

WorkOrderLineItemHistory

Represents the history of changes made to tracked fields on a work order line item. This object is available in API version 36.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Work orders or Field Service Lightning must be enabled in your organization, and field tracking for work order line item fields must be configured.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p>

Field Name	Details
	<p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
WorkOrderLineItemId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the work order line item being tracked. The history is displayed on the detail page for this record.</p>

WorkOrderShare

Represents a sharing entry on a work order. This object is available in API version 36.0 and later.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Special Access Rules

Work orders or Field Service Lightning must be enabled in your organization. Community users can't access this object.

Fields

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>Level of access that the user or group has to the work order. The possible values are:</p> <ul style="list-style-type: none"> • <i>Read</i> • <i>Edit</i> • <i>All</i> (This value isn't valid for create or update calls.) <p>Set to an access level that is at least equal to the organization's default work order access level.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description</p> <p>The work order associated with the sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description</p> <p>The reason why this sharing entry exists. You can write to this field only when its value is omitted or set to <i>Manual</i> (default). Valid values include:</p>
	<ul style="list-style-type: none"> • <i>Manual</i>—The user or group has access because a user with "All" access manually shared the work order. • <i>Owner</i>—The user is the owner of the work order. • <i>Team</i>—The user or group has team access. • <i>Rule</i>—The user or group has access via a work order sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description</p> <p>(Read Only) ID of the user or group that has access to the work order.</p>

WorkPerformanceCycle

Represents feedback that is gathered to assess the performance of a specific set of employees.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
ActivityFrom	<p>Type date</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The date that you want to start filtering the work.com objects to help requesters create accurate summaries. The start of the evaluation period.</p>
ActivityTo	<p>Type date</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The date that you want to stop filtering the work.com objects to help requesters create accurate summaries. The end of the evaluation period.</p>
CurrentTask	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The current task that the performance summary cycle is engaged in, including deploying and sharing.</p>
LastManagerRequestsSharedDate	<p>Type dateTime</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The date when all manager requests are set to be shared.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	Description
	The time stamp that indicates when the current user last viewed a record that is related to this WorkPerformanceCycle.
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed this WorkPerformanceCycle. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.
Name	Type
	string
	Properties
	Create, Filter, Group, idLookup, Sort, Update
	Description
	The name of the performance summary cycle that employees will participate in. This name is created by the administrator and is visible on all respective notifications and in the UI.
OwnerId	Type
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	ID of the owner of the WorkPerformanceCycle.
State	Type
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	The state that the performance summary cycle is in. Available pick list values:
	<ul style="list-style-type: none"> • Setup: The summary is in draft. • In Progress: The summary is deployed and people are answering the questions that were created. • Finished: The summary is no longer in progress. • Error: The summary encountered an error.

WorkPerformanceCycleFeed

Represents a single feed item in the feed that is displayed on a Work.com Performance detail page.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The content of the FeedItem. Required when <code>Type</code> is <code>TextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>. This field is the message that appears in the feed.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedComments associated with this feed item.</p>
ContentData	<p>Type base64</p> <p>Properties Nillable</p> <p>Description Available in API version 36.0 and earlier only. Required if <code>Type</code> is <code>ContentPost</code>. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentDescription	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. The description of the file specified in <code>ContentData</code>.</p>

Field Name	Details
ContentFileName	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. This field is required if <code>Type</code> is <code>ContentPost</code>. The name of the file uploaded to the feed. Setting <code>ContentFileName</code> automatically sets <code>Type</code> to <code>ContentPost</code>.</p>
ContentSize	<p>Type int</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The size of the file (in bytes) uploaded to the feed.</p>
ContentType	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description Available in API version 36.0 and earlier only. Read only. Automatically determined during insert. The MIME type of the file uploaded to the feed.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if a client application migrates multiple posts and comments from another application into a feed, then <code>InsertedById</code> is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the feed item <code>Body</code> contains rich text. Set <code>IsRichText</code> to <code>true</code> if you post a rich text feed item via the SOAP API. Otherwise, the post is rendered as plain text.</p>

Field Name	Details
	<p>Rich text supports the following HTML tags:</p> <ul style="list-style-type: none"> • <p> <p> Tip: Though the
 tag isn't supported, you can use <p>&nbsp;</p> to create lines.</p> <ul style="list-style-type: none"> • • <code> • <i> • <u> • <s> • • • • <p>The tag is accessible only via the API and must reference files in Salesforce similar to this example: </p> <p> Note: In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of FeedLikes associated with this feed item.</p>
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a LinkPost.</p>
NetworkScope	<p>Type picklist</p> <p>Properties Group, Nillable, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>Specifies whether this feed item is available in the default community, a specific community, or all communities. If Salesforce Communities is enabled for your organization, this field is available in API version 30.0 and later.</p> <p><code>NetworkScope</code> can have the following values:</p> <ul style="list-style-type: none"> • <code>NetworkId</code>—The ID of the community in which the FeedItem is available. If left empty, the feed item is only available in the default community. • <code>AllNetworks</code>—The feed item is available in all communities. <p>Note the following exceptions for <code>NetworkScope</code>:</p> <ul style="list-style-type: none"> • Only feed items with a Group or User parent can set a <code>NetworkId</code> or a null value for <code>NetworkScope</code>. • For feed items with a record parent, users can set <code>NetworkScope</code> only to <code>AllNetworks</code>. • You can't filter a FeedItem on the <code>NetworkScope</code> field.
ParentId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the group that is tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>ID of the ContentVersion object associated with a ContentPost. This field is null for all posts except ContentPost.</p> <p>For example, set this field to an existing ContentVersion and post it to a feed as a FeedItem object of Type ContentPost.</p>
Title	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The title of the FeedItem. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL, and this field is the link name.</p>

Field Name	Details
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for ContentPost, LinkPost, and TextPost, don't create feed item types directly from the API</p> <ul style="list-style-type: none"> • ActivityEvent—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). <p>For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.</p> <ul style="list-style-type: none"> • AdvancedTextPost—created when a user posts a group announcement. • AnnouncementPost—Not used. • ApprovalPost—generated when a user submits an approval. • BasicTemplateFeedItem—Not used. • CanvasPost—a post made by a canvas app posted on a feed. • CollaborationGroupCreated—generated when a user creates a public group. • CollaborationGroupUnarchived—Not used. • ContentPost—a post with an attached file. • CreatedRecordEvent—generated when a user creates a record from the publisher. • DashboardComponentAlert—generated when a dashboard metric or gauge exceeds a user-defined threshold. • DashboardComponentSnapshot—created when a user posts a dashboard snapshot on a feed. • LinkPost—a post with an attached URL. • PollPost—a poll posted on a feed. • ProfileSkillPost—generated when a skill is added to a user's Chatter profile. • QuestionPost—generated when a user posts a question. • ReplyPost—generated when Chatter Answers posts a reply. • RipplePost—generated when a user creates a Thanks badge in Work.com. • TextPost—a direct text entry on a feed. • TrackedChange—a change or group of changes to a tracked field.

Field Name	Details
	<ul style="list-style-type: none"> UserStatus—automatically generated when a user adds a post. Deprecated. <p>The following values appear in the Type picklist for all feed objects but apply only to CaseFeed:</p> <ul style="list-style-type: none"> AttachArticleEvent—generated event when a user attaches an article to a case. CallLogPost—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. CaseCommentPost—generated event when a user adds a case comment for a case object. ChangeStatusPost—generated event when a user changes the status of a case. ChatTranscriptPost—generated event when Live Agent transcript is saved to a case. EmailMessageEvent—generated event when an email related to a case object is sent or received. FacebookPost—generated when a Facebook post is created from a case. Deprecated. MilestoneEvent—generated when a case milestone is completed or reaches violation status. SocialPost—generated when a social post is created from a case.
Visibility	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The visibility of a FeedItem: <ul style="list-style-type: none"> AllUsers: visible to all users InternalUsers: visible to internal users </p>

WorkPerformanceCycleHistory

Represents the history of changes to the values in the fields of a WorkPerformanceCycle object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed.</p>
WorkPerformanceCycleId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the WorkPerformanceCycle.</p>

WorkPerformanceCycleShare

Represents a sharing entry on a WorkPerformanceCycle object.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The user's or group's level of access to the work performance cycle. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records. <p>This field must be set to an access level that is higher than the organization's default access level for work performance cycles.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the WorkPerformanceCycle object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none"> • Owner: User is the owner of the WorkPerformanceCycle or is in a user role above the WorkPerformanceCycle owner in the role hierarchy. • Manual: User or group has access, because a user with "All" access manually shared the WorkPerformanceCycle with the user or group. • Rule: User or group has access via a WorkPerformanceCycle sharing rule.

Field Name	Details
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the user or group that was given access to the performance cycle share. This field can't be updated.</p>

WorkReward

Used to store reward codes tied to a Reward Fund. Reward Funds must have at least one WorkReward record.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

You must have the Reward permission enabled in order to use the Rewards feature, including WorkRewardFund and WorkReward.

Additional Considerations and Related Objects

WorkReward is a lookup to WorkRewardFund. WorkRewardFund must have at least one WorkReward record to be available for use. Each WorkBadge record with a `RewardId` indicates a reward badge given to a Recipient.

Fields

Field Name	Details
Code	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Represents a single reward code tied to a RewardFundId.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	<p>Description Represents the User ID of Owner of WorkReward record</p>
RecipientId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description Salesforce User ID for User associated with this WorkReward record.</p>
RedemptionDisclaimer	<p>Type textarea</p> <p>Properties Nullable</p> <p>Description The disclaimer information about the WorkReward.</p>
RedemptionInfo	<p>Type textarea</p> <p>Properties Nullable</p> <p>Description The instructions for redeeming the WorkReward.</p>
RedemptionUrl	<p>Type textarea</p> <p>Properties Nullable</p> <p>Description The URL for redeeming the WorkReward.</p>
RewardFundId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Salesforce unique ID for WorkRewardFund record that is associated with WorkReward record.</p>
RewardFundTypeId	<p>Type reference</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Salesforce unique ID of the WorkRewardFundType associated with the WorkReward.</p>
Value	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The value of the WorkReward.</p>

WorkRewardFund

Represents a Reward Fund and describes the Reward Fund attributes.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

To use the Rewards feature, including WorkRewardFund and WorkReward, you must have the Reward permission enabled. To create Rewards, the user must have Create on WorkRewardFund, which is not a standard permission.

Additional Considerations and Related Objects

WorkReward is a lookup to WorkRewardFund. WorkRewardFund must have at least one WorkReward record available. Each WorkBadgeDefinition with a RewardFundId is a “Reward Badge.”

Fields

Field Name	Details
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description
	Indicates whether the WorkRewardFund is active (<code>true</code>) or not (<code>false</code>).
LastReferencedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed a record that is related to this WorkRewardFund.
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed this WorkRewardFund. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.
Name	Type
	string
	Properties
	Create, Filter, Group, idLookup, Sort, Update
	Description
	Required. Name of the Reward Fund.
OwnerId	Type
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Salesforce unique ID of User who is the Owner of the WorkRewardFund record.
RewardFundTypeId	Type
	reference
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Salesforce unique ID of the WorkRewardFundType that is associated with the WorkRewardFund.

Field Name	Details
TotalCodeCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total reward codes that are available in the WorkRewardFund. Derived from WorkReward records that are associated with the WorkRewardFund.</p>
Type	<p>Type string</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description RewardType of the WorkRewardFund. Default is Amazon.com.</p>
UsedCodeCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total reward codes that are used in the WorkRewardFund. Derived from the total assigned WorkReward records that are associated with the WorkRewardFund.</p>
Value	<p>Type currency</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description Value of each of the reward codes in the WorkRewardFund.</p>

WorkRewardFundHistory

Represents the history of changes to the values in the fields of a WorkRewardFund object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Type of change (e.g., created).</p>
NewValue	<p>Type Any Type</p> <p>Properties Nillable, Sort</p> <p>Description Updated value of record.</p>
OldValue	<p>Type Any Type</p> <p>Properties Nillable, Sort</p> <p>Description Previous value of record.</p>
WorkRewardFundId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of WorkRewardFund record.</p>

WorkRewardFundShare

Share records for WorkRewardFund.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description CRUD Access Level (picklist values: Read Only, Read/Write, Owner).</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID for WorkRewardFund record.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Owner or Manual sharing. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description User or Group ID for WorkRewardFund record.</p>

WorkRewardFundType

Represents the type of WorkRewardFund object.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field Name	Details
CreditSystem	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The credit system that is used by the WorkRewardFundType object (gift codes or points). If points are selected, the reward message will not consider the CurrencyCode field.</p>
CurrencyCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The currency code of the WorkRewardFundType</p>
IsActive	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Whether the WorkRewardFundType is active and available in the UI</p>
IsPredefined	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Whether the WorkRewardFundType is predefined (<code>true</code>) or not (<code>false</code>)</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	Description
	The time stamp that indicates when the current user last viewed a record that is related to this WorkRewardFundType.
LastViewedDate	Type
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The time stamp that indicates when the current user last viewed this WorkRewardFundType. If this value is null, this record might have been only referenced (<code>LastReferencedDate</code>) and not viewed.
Name	Type
	string
	Properties
	Create, Filter, Group, idLookup, Sort, Update
	Description
	The name of the WorkRewardFundType
OwnerId	Type
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	The ID of the WorkRewardFundType owner
RedemptionDisclaimer	Type
	textarea
	Properties
	Create, Nillable, Update
	Description
	The redemption disclaimer text for the WorkRewardFundType
RedemptionInfo	Type
	textarea
	Properties
	Create, Nillable, Update
	Description
	Redemption text for the WorkRewardFundType

Field Name	Details
RedemptionUrl	<p>Type textarea</p> <p>Properties Create, Nullable, Update</p> <p>Description The URL that's linked to the redemption</p>
UploadCodeColumn	<p>Type int</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The column where the reward code is contained in the CSV file. The upload uses the second value by default.</p>
UploadValueColumn	<p>Type int</p> <p>Properties Create, Filter, Group, Nullable, Sort, Update</p> <p>Description The column where the reward value is contained in the CSV file. The upload uses the third column by default.</p>

WorkRewardFundTypeHistory

Represents the history of changes to the values in the fields of a WorkRewardFundType object.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description The name of the field that was changed</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the changed field</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed</p>
WorkRewardFundTypeID	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the WorkRewardFundType</p>

WorkRewardFundTypeShare

Represents a sharing entry on a WorkRewardFundType.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p>
	<p>Description The user's or group's level of access to the work reward fund type. The possible values are:</p> <ul style="list-style-type: none"> • Read • Edit • All: This value is not valid when you create, update, or delete records. <p>This field must be set to an access level that is higher than the organization's default access level for WorkRewardFundType objects.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the WorkRewardFundType object that is associated with this sharing entry.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The reason that this sharing entry exists. Read-only. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings. Valid values include:</p> <ul style="list-style-type: none"> • Owner: User is the owner of the WorkRewardFundType or is in a user role above the WorkRewardFundType owner in the role hierarchy. • Manual: User or group has access, because a user with "All" access manually shared the WorkRewardFundType with the user or group. • Rule: User or group has access via a WorkRewardFundType sharing rule.
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user or group that was given access to the WorkRewardFundType. This field can't be updated.</p>

WorkRewardHistory

Represents the history of changes to the fields of a WorkReward.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The latest value of the field before it was changed</p>
WorkRewardId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the WorkReward</p>

WorkRewardShare

Share records for WorkReward object.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description CRUD Access Level (picklist values: Read Only, Read/Write, Owner).</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID for WorkReward record.</p>
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Owner or Manual sharing. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description User or Group ID for WorkReward record.</p>

WorkThanks

Represents the source and message of a thanks post.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), undelete(), update(), upsert()`

Additional Considerations and Related Objects

WorkBadge is a lookup to WorkThanks. Each WorkBadge record must derive a Sourceld from WorkThanks.

Fields

Field Name	Details
FeedItemId	Type reference Properties Filter, Group, Nullable, Sort Description ID of the FeedItem related to the thanks badge.
GiverId	Type reference Properties Create, Filter, Group, Sort Description Salesforce user ID for the giver of the Thanks record.
Message	Type textarea Properties Create Description Required. Message associated with the Thanks record.
NetworkId	Type reference Properties Create, Filter, Group, Nullable, Sort, Update

Field Name	Details
	<p>Description The ID of the community that this WorkThanks is associated with. This field is available only if Salesforce Communities is enabled in your organization.</p>
OwnerId	<p>Type reference</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Salesforce user ID for the owner of the badge record (typically the same user as the giver of the record).</p>

WorkThanksShare

Share records for WorkThanks object.

Supported Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field Name	Details
AccessLevel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description CRUD Access Level (picklist values: Read Only, Read/Write, Owner).</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID for WorkThanks record.</p>

Field Name	Details
RowCause	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Owner or Manual sharing. You can create a value for this field in API versions 32.0 and later with the correct organization-wide sharing settings.</p>
UserOrGroupId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description User or Group ID for WorkThanks record.</p>

WorkType

Represents a type of work to be performed. Work types are templates that can be applied to work order or work order line items. This object is available in API version 38.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(),
undelete(), update(), upsert()
```

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Description	<p>Type textarea</p> <p>Properties Create, Nillable, Update</p> <p>Description The description of the work type. Try to add details about the task or tasks that this work type represents.</p>

Field Name	Details
DurationType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Defaulted on create, Restricted picklist, Sort, Update</p> <p>Description The unit of the Estimated Duration: Minutes or Hours.</p>
EstimatedDuration	<p>Type double</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The estimated length of the work. The estimated duration is in minutes or hours based on the value selected in the Duration Type field.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the work type was last modified. Its label in the user interface is Last Modified Date.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the work type was last viewed by the current user.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The name of the work type. Try to use a name that helps users quickly understand the type of work orders that can be created from the work type. For example, "Annual Refrigerator Maintenance" or "Valve Replacement."</p>
ShouldAutoCreateSvcAppt	<p>Type boolean</p>

Field Name	Details
	<p>Properties Create, Filter, Group, Defaulted on create, Sort, Update</p>
	<p>Description Select this option to have a service appointment automatically created on work orders and work order line items that use the work type.</p> <p> Note: By default, the Due Date on auto-created service appointments is seven days after the created date. Admins can adjust this offset from the Field Service Settings page in Setup.</p>

Usage

To apply a work type to a work order or work order line item, select the work type in the `Work Type` lookup field on the record. When you add a work type, the work order or work order line item inherits the work type's duration values and required skills.

 **Note:**

- If needed, you can update the duration values and required skills on a work order or work order line item after they're inherited from the work type.
- If a work order or work order line item already has required skills, associating it with a work type doesn't cause it to inherit the work type's required skills.
- Customizations to required skills, such as validation rules or Apex triggers, are not carried over from work types to work orders and work order line items.

WorkTypeFeed

Represents a single feed item on a work type record detail page. This object is available in API version 38.0 and later.

A work type feed shows changes to tracked fields on a work type record. It also shows posts and comments about the record. It's a useful way to stay up to date with changes made to work types in Salesforce.

Supported Calls

```
delete(), getDeleted(), getUpdated(), query(), retrieve()
```

Special Access Rules

Field Service Lightning must be enabled in your organization, and you can delete all feed items you created. To delete feed items you didn't create, you must have one of these permissions:

- "Modify All Data"
- "Modify All" on the Operating Hours object
- "Moderate Chatter"

Fields

Field Name	Details
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The message that appears in the feed. Required when Type is TextPost. Optional when Type is ContentPost or LinkPost.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the user who added this object to the feed. For example, if an application migrates posts and comments from another application into a feed, the value is set to the ID of the logged-in user.</p>
IsRichText	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description <code>true</code> indicates that the body of the feed contains rich text. <code>false</code> indicates that the post is rendered as plain text.</p>
LikeCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of feed likes associated with this feed item.</p>

Field Name	Details
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a link post.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the work type record tracked in the feed. The feed is displayed on the detail page for this record.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the ContentVersion object associated with a content post. This field is null for all posts, except ContentPost.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the Type is LinkPost, the Link URL is the URL, and this field is the link name.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item.</p>

WorkTypeHistory

Represents the history of changes made to tracked fields on a work type. This object is available in API version 38.0 and later.

Supported Calls

`describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

Field Service Lightning must be enabled.

Fields

Field Name	Details
Field	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The name of the field that was changed.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The value of the field before it was changed.</p>
WorkTypeId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>

Field Name	Details
	Description
	ID of the work type being tracked. The history is displayed on the detail page for this record.

Reusable Terms for API Objects

API Version Availability.

This object is available in API version 38.0 and later.

This object is available in API version 37.0 and later.

This object is available in API version 27.0 and later.

This object is available in API version 21.0 and later.

This object is available in API version 20.0 and later.

This object is available in API version 18.0 and later.

API Reusables



Note: To programmatically update owner sharing rules, we recommend that you use Metadata API. Contact Salesforce customer support to enable access to this object for your org.

CHAPTER 9 Core Calls

The following table lists supported calls in the API in alphabetical order, and provides a brief description for each. Click a call name to see syntax, usage, and more information for that call.

 **Note:** For a list of API utility calls, see [Utility Calls](#), and for a list of describe calls, see [Describe Calls](#).

Call	Description
convertLead()	Converts a Lead into an Account , Contact , or (optionally) an Opportunity .
create()	Adds one or more new individual objects to your organization's data.
delete()	Deletes one or more individual objects from your organization's data.
emptyRecycleBin()	Delete records from the recycle bin immediately.
findDuplicates() on page 2168	Performs rule-based searches for duplicate records. The input is an array of sObject , each of which specifies the values to search for and the type of object that supplies the duplicate rules. The output identifies the detected duplicates for each object that supplies the duplicate rules. findDuplicates() applies the rules to the values to do the search. The output identifies the detected duplicates for each sObject .
getDeleted()	Retrieves the IDs of individual objects of the specified object that have been deleted since the specified time. For information on IDs, see ID Field Type .
getUpdated()	Retrieves the IDs of individual objects of the specified object that have been updated since the specified time. For information on IDs, see ID Field Type .
invalidateSessions()	Ends one or more sessions specified by sessionId .
login()	Logs in to the login server and starts a client session.
logout()	Ends the session of the logged-in user.
merge()	Merges records of the same object type.
process()	Submits an array of approval process instances for approval, or processes an array of approval process instances to be approved, rejected, or removed.
query()	Executes a query against the specified object and returns data that matches the specified criteria.
queryAll()	Same as query() , but includes deleted and archived items.
queryMore()	Retrieves the next batch of objects from a query.
retrieve()	Retrieves one or more objects based on the specified object IDs.

Call	Description
<code>search()</code>	Executes a text search in your organization's data.
<code>undelete()</code>	Undelete records identified with <code>queryAll()</code> .
<code>update()</code>	Updates one or more existing objects in your organization's data.
<code>upsert()</code>	Creates new objects and updates existing objects; matches on a custom field to determine the presence of existing objects.

Samples

The samples in this section are based on the enterprise WSDL file. They assume that you have already imported the WSDL file and created a connection. To learn how to do so, see the [Quick Start](#) tutorial.

convertLead()

Converts a [Lead](#) into an [Account](#), [Contact](#), or (optionally) an [Opportunity](#).

Syntax

```
LeadConvertResult[] = connection.convertLead(leadConverts LeadConvert[]);
```

Usage

Use `convertLead()` to convert a [Lead](#) into an [Account](#) and [Contact](#), as well as (optionally) an [Opportunity](#). To convert a [Lead](#), your client application must be logged in with the “Convert Leads” permission and the “Edit” permission on leads, as well as “Create” and “Edit” on the [Account](#), [Contact](#), and [Opportunity](#) objects.

This call provides an easy way to convert the information in a qualified lead to a new or updated account, contact, and opportunity. Your organization can set its own guidelines for determining when a lead is qualified, but typically, a lead can be converted as soon as it becomes a real opportunity that you want to forecast.

If data is merged into existing account and contact objects, then only empty fields in the target object are overwritten—existing data (including IDs) are not overwritten. The only exception to this is if your client application sets `overwriteLeadSource` to `true`, in which case the `LeadSource` field in the target [Contact](#) object will be overwritten with the contents of the `LeadSource` field in the source [Lead](#) object.

When converting leads, consider the following rules and guidelines:

Field Mappings

The system automatically maps standard lead fields to standard account, contact, and opportunity fields. For custom lead fields, your Salesforce administrator can specify how they map to custom account, contact, and opportunity fields.

Record Types

If the organization uses record types, the default record type of the new owner is assigned to records created during lead conversion. For more information about record types, see the Salesforce online help.

Picklist Values

The system assigns the default picklist values for the account, contact, and opportunity when mapping any standard lead picklist fields that are blank. If your organization uses record types, blank values are replaced with the default picklist values of the new record owner.

String Values

Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the [AllowFieldTruncationHeader](#) SOAP header.

Errors

If any of the leads fail to convert as part of a bulk operation, the lead conversion is retried for each lead individually.

Automatic Subscriptions for Chatter Feeds

When you convert a lead into a new account, contact, and opportunity, the lead owner is unsubscribed from the lead account. The lead owner, the owner of the generated records, and users that were subscribed to the lead aren't automatically subscribed to the generated records, unless they have automatic subscriptions enabled in their Chatter feed settings. They must have automatic subscriptions enabled to see changes to the account, contact, and opportunity records in their news feed.

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later.

Basic Steps for Converting Leads

Converting leads involves the following basic steps:

1. The client application determines the IDs of any lead(s) to be converted.
2. Optionally, the client application determines the IDs of any account(s) to merge the lead into. The client application can use SOSL or SOQL to search for accounts that match the lead name, as in the following example:

```
select id, name from account where name='CompanyNameOfLeadBeingMerged'
```

3. Optionally, the client application determines the IDs of contact(s) to merge the lead into. The client application can use SOSL or SOQL to search for contacts that match the lead contact name, as in the following example:

```
select id, name from contact where firstName='FirstName' and lastName='LastName' and accountID = '001...'
```

4. Optionally, the client application determines whether opportunities should be created from the leads.

5. The client application queries the LeadStatus table to obtain all of the possible converted status options (

```
SELECT Id, MasterLabel FROM LeadStatus WHERE IsConverted=true
```

), and then selects a value for the Converted Status.

6. The client application calls `convertLead()`.
7. The client application iterates through the returned result(s) and examine each `LeadConvertResult` object to determine whether conversion succeeded for each lead.
8. As an optional best practice, the client application creates tasks in which the `WhoId` is the `ContactId` and, if an opportunity is created, the `WhatId` is the `OpportunityId`.
9. Optionally, when converting leads owned by a queue, the owner must be specified. This is because accounts and contacts cannot be owned by a queue. Even if you are specifying an existing account or contact, you must still specify an owner.

Sample Code—Java

This sample shows how to convert leads. It creates two leads and converts them. Next, it iterates through the lead conversion results and writes the IDs of the account, contact, and opportunity created for each lead.

```
public String[] convertLeadRecords() {
    String[] result = new String[4];
    try {

        // Create two leads to convert
        Lead[] leads = new Lead[2];
        Lead lead = new Lead();
        lead.setLastName("Mallard");
        lead.setFirstName("Jay");
        lead.setCompany("Wingo Ducks");
        lead.setPhone("(707) 555-0328");
        leads[0] = lead;
        lead = new Lead();
        lead.setLastName("Platypus");
        lead.setFirstName("Ogden");
        lead.setCompany("Denio Water Co.");
        lead.setPhone("(775) 555-1245");
        leads[1] = lead;
        SaveResult[] saveResults = connection.create(leads);

        // Create a LeadConvert array to be used
        // in the convertLead() call
        LeadConvert[] leadsToConvert = new LeadConvert[saveResults.length];

        for (int i = 0; i < saveResults.length; ++i) {
            if (saveResults[i].isSuccess()) {
                System.out
                    .println("Created new Lead: " + saveResults[i].getId());
                leadsToConvert[i] = new LeadConvert();
                leadsToConvert[i].setConvertedStatus("Closed - Converted");
                leadsToConvert[i].setLeadId(saveResults[i].getId());
                result[0] = saveResults[i].getId();
            } else {
                System.out.println("\nError creating new Lead: "

```

```
        + saveResults[i].getErrors()[0].getMessage());
    }
}

// Convert the leads and iterate through the results
LeadConvertResult[] lcResults = connection.convertLead(leadsToConvert);
for (int j = 0; j < lcResults.length; ++j) {
    if (lcResults[j].isSuccess()) {
        System.out.println("Lead converted successfully!");
        System.out.println("Account ID: " + lcResults[j].getAccountId());
        System.out.println("Contact ID: " + lcResults[j].getContactId());
        System.out.println("Opportunity ID: "
            + lcResults[j].getOpportunityId());
    } else {
        System.out.println("\nError converting new Lead: "
            + lcResults[j].getErrors()[0].getMessage());
    }
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
return result;
}
```

Sample Code—C#

This sample shows how to convert leads. It creates two leads and converts them. Next, it iterates through the lead conversion results and writes the IDs of the account, contact, and opportunity created for each lead.

```
public String[] convertLeadRecords()
{
    String[] result = new String[4];
    try
    {
        // Create two leads to convert
        Lead[] leads = new Lead[2];
        Lead lead = new Lead();
        lead.LastName = "Mallard";
        lead.FirstName = "Jay";
        lead.Company = "Wingo Ducks";
        lead.Phone = "(707) 555-0328";
        leads[0] = lead;
        lead = new Lead();
        lead.LastName = "Platypus";
        lead.FirstName = "Ogden";
        lead.Company = "Denio Water Co.";
        lead.Phone = "(775) 555-1245";
        leads[1] = lead;
        SaveResult[] saveResults = binding.create(leads);

        // Create a LeadConvert array to be used
        // in the convertLead() call
        LeadConvert[] leadsToConvert =
            new LeadConvert[saveResults.Length];
```

```
for (int i = 0; i < saveResults.Length; ++i)
{
    if (saveResults[i].success)
    {
        Console.WriteLine("Created new Lead: " +
            saveResults[i].id);
        leadsToConvert[i] = new LeadConvert();
        leadsToConvert[i].convertedStatus = "Closed - Converted";
        leadsToConvert[i].leadId = saveResults[i].id;
        result[0] = saveResults[i].id;
    }
    else
    {
        Console.WriteLine("\nError creating new Lead: " +
            saveResults[i].errors[0].message);
    }
}
// Convert the leads and iterate through the results
LeadConvertResult[] lcResults =
    binding.convertLead(leadsToConvert);
for (int j = 0; j < lcResults.Length; ++j)
{
    if (lcResults[j].success)
    {
        Console.WriteLine("Lead converted successfully!");
        Console.WriteLine("Account ID: " +
            lcResults[j].accountId);
        Console.WriteLine("Contact ID: " +
            lcResults[j].contactId);
        Console.WriteLine("Opportunity ID: " +
            lcResults[j].opportunityId);
    }
    else
    {
        Console.WriteLine("\nError converting new Lead: " +
            lcResults[j].errors[0].message);
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
return result;
}
```

LeadConvert Arguments

This call accepts an array of LeadConvert objects (100 maximum). A LeadConvert object contains the following properties.

Name	Type	Description
accountId	ID	ID of the Account into which the lead will be merged. Required only when updating an existing account, including person accounts. If no <code>accountId</code> is specified, then the API creates a new account. To create a new account, the client application must be logged in with sufficient access rights. To merge a lead into an existing account, the client application must be logged in with read/write access to the specified account. The account name and other existing data are not overwritten. For information on IDs, see ID Field Type .
contactId	ID	ID of the Contact into which the lead will be merged (this contact must be associated with the specified <code>accountId</code> , and an <code>accountId</code> must be specified). Required only when updating an existing contact.
		<p>Important: If you are converting a lead into a person account, do not specify the <code>contactId</code> or an error will result. Specify only the <code>accountId</code> of the person account.</p> <p>If no <code>contactId</code> is specified, then the API creates a new contact that is implicitly associated with the Account. To create a new contact, the client application must be logged in with sufficient access rights. To merge a lead into an existing contact, the client application must be logged in with read/write access to the specified contact. The contact name and other existing data are not overwritten (unless <code>overwriteLeadSource</code> is set to <code>true</code>, in which case only the <code>LeadSource</code> field is overwritten).</p>
convertedStatus	string	Valid LeadStatus value for a converted lead. Required. To obtain the list of possible values, the client application queries the LeadStatus object. For example:
		<pre>SELECT Id, MasterLabel FROM LeadStatus WHERE IsConverted=true</pre>
doNotCreateOpportunity	boolean	Specifies whether to create an Opportunity during lead conversion (<code>false</code> , the default) or not (<code>true</code>). Set this flag to <code>true</code> only if you do not want to create an opportunity from the lead. An opportunity is created by default.
leadId	ID	ID of the Lead to convert. Required. For information on IDs, see ID Field Type .
opportunityName	string	Name of the opportunity to create. If no name is specified, then this value defaults to the company name of the lead. The maximum length of this field is 80 characters. If <code>doNotCreateOpportunity</code> argument is <code>true</code> , then no Opportunity is created and this field must be left blank; otherwise, an error is returned.
overwriteLeadSource	boolean	Specifies whether to overwrite the <code>LeadSource</code> field on the target Contact object with the contents of the <code>LeadSource</code> field in the source Lead object (<code>true</code> , or not (<code>false</code> , the default)). To set this field to <code>true</code> , the client application must specify a <code>contactId</code> for the target contact.
ownerId	ID	Specifies the ID of the person to own any newly created account, contact, and opportunity. If the client application does not specify this value, then

Name	Type	Description
		the owner of the new object will be the owner of the lead. Not applicable when merging with existing objects—if an ownerId is specified, the API does not overwrite the ownerId field in an existing account or contact. For information on IDs, see ID Field Type .
sendNotificationEmail	boolean	Specifies whether to send a notification email to the owner specified in the ownerId (true) or not (false, the default).

Response

[LeadConvertResult\[\]](#)

Fault

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

LeadConvertResult

This call returns an array of LeadConvertResult objects. Each element in the LeadConvertResult array corresponds to the LeadConvert[] array passed as the leadConverts parameter in the [convertLead\(\)](#) call. For example, the object returned in the first index in the LeadConvertResult array matches the object specified in the first index of the LeadConvert[] array. A LeadConvertResult object has the following properties:

Name	Type	Description
accountId	ID	ID of the new Account (if a new account was specified) or the ID of the account specified when convertLead() was invoked.
contactId	ID	ID of the new Contact (if a new contact was specified) or the ID of the contact specified when convertLead() was invoked. For information on IDs, see ID Field Type .
leadId	ID	ID of the converted Lead . For information on IDs, see ID Field Type .
opportunityId	ID	ID of the new Opportunity , if one was created when convertLead() was invoked. For information on IDs, see ID Field Type .
success	boolean	Indicates whether the convertLead() call succeeded (true) or not (false) for this object.
errors	Error[]	If an error occurred during the create() call, an array of one or more Error objects providing the error code and description.

create()

Adds one or more new records to your organization's data.

Syntax

```
SaveResult[] = connection.create(sObject[] sObjects);
```

Usage

Use `create()` to add one or more records, such as an [Account](#) or [Contact](#) record, to your organization's information. The `create()` call is analogous to the INSERT statement in SQL.

When creating objects, consider the following rules and guidelines.

Permissions

Your client application must be logged in with sufficient access rights to create records within the specified object. For more information, see [Factors that Affect Data Access](#).

Special Handling

Certain objects—and certain fields within those objects—require special handling or permissions. For example, you might also need permissions to access the object's parent object. Before you attempt to `create()` a record for a particular object, be sure to read its description in the [Standard Objects](#).

Createable Fields

Only objects where `createable` is `true` can be created via the `create()` call. To determine whether a given object can be created, your client application can invoke the `describeSObjects()` call on the object and inspect its `createable` property.

Automatically Maintained Fields

The API generates unique values for ID fields automatically. For `create()`, you cannot explicitly specify an ID value in the `sObject`. The `SaveResult` object contains the ID of each record that was successfully created. For information on IDs, see [ID Field Type](#).

The API populates certain fields automatically, such as `CreatedDate`, `CreatedById`, `LastModifiedDate`, `LastModifiedById`, and `SystemModstamp`. You cannot explicitly specify these values.

Required Fields

For required fields that do not have a preconfigured default value, you must supply a value. For more information, see [Required Fields](#).

Default Values

For some objects, some fields have a default value, such as `OwnerId`. If you do not specify a value for such fields, the API populates the fields with the default value. For example, if you do not override `OwnerId`, then the API populates this field with the user ID associated with the user as whom your client application is logged in.

- For required fields that do not have a preconfigured default value, you must supply a value.
- For all other fields in the object, if you do not explicitly specify a value, then its value is `null` (`VT_EMPTY`).

Referential Integrity

Your client application must conform to the rules of referential integrity. For example, if you are creating a record for an object that is the child of a parent object, you must supply the foreign key information that links the child to the parent. For example, when creating a [CaseComment](#), you must supply the valid case ID for the parent [Case](#), and that parent Case must exist in the database.

Valid Data Values

You must supply values that are valid for the field's data type, such as integers (not alphabetic characters) for integer fields. In your client application, follow the data formatting rules specified for your programming language and development tool (your development tool will handle the appropriate mapping of data types in SOAP messages).

String Values

When storing values in string fields, the API trims any leading and trailing whitespace. For example, if the value of a name field is entered as " ABC Company ", then the value is stored in the database as "ABC Company".

Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the [AllowFieldTruncationHeader](#) SOAP header.

Assignment Rules

When creating new [Account](#) (accounts fire Territory Management assignment rules), [Case](#), or [Lead](#) records, your client application can set options in the [AssignmentRuleHeader](#) to have the case or lead automatically assigned to one or more users based on assignment rules configured in the Salesforce user interface.

Maximum Number of Records Created

Your client application can add up to 200 records in a single `create()` call. If a create request exceeds 200 objects, then the entire operation fails.

Rollback on Error

The [AllOrNoneHeader](#) header allows you to roll back all changes unless all records are processed successfully. This header is available in API version 20.0 and later. Allows a call to roll back all changes unless all records are processed successfully.

Automatic Subscriptions for Chatter Feeds

To subscribe to records they create, users must enable the `Automatically follow records that I create` option in their personal settings. If users have automatic subscriptions enabled, they automatically follow the records they create and see changes to those records in their Chatter feed on the Home tab.

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later. The `EntitySubscription` object represents a subscription of a user following a record or another user.

Disabling Feed Notifications

If you're processing a large number of records and don't want to track the changes in various feeds related to the records, use `DisableFeedTrackingHeader`. This is especially useful for bulk changes.

Creating Records for Different Object Types

You can create records for multiple object types, including custom objects, in one call with API version 20.0 and later. For example, you could create a contact and an account in one call. You can create records for up to 10 object types in one call.

Records are saved in the same order that they are entered in the `sObjects` input array. If you are entering new records that have a parent-child relationship, the parent record must precede the child record in the `sObjects` array. For example, if you are creating a contact that references an account that is also being created in the same call, the account must have a smaller index in the `sObjects` array than the contact does. The contact references the account by using an `External ID` field.

You can't add a record that references another record of the same object type in the same call. For example, the `Contact` object has a `Reports To` field that is a reference to another contact. You can't create two contacts in one call if one contact uses the `Reports To` field to reference a second contact in the `sObjects` array. You can create a contact that references another contact that has been previously created.

Records for different object types are broken into multiple chunks by Salesforce. A chunk is a subset of the `sObjects` input array and each chunk contains records of one object type. Data is committed on a chunk-by-chunk basis. Any Apex triggers related to the records in a chunk are invoked once per chunk. Consider an `sObjects` input array containing the following set of records:

```
account1, account2, contact1, contact2, contact3, case1, account3, account4, contact4
```

Salesforce splits the records into five chunks:

1. account1, account2
2. contact1, contact2, contact3
3. case1
4. account3, account4
5. contact4

Each call can process up to 10 chunks. If the `sObjects` array contains more than 10 chunks, you must process the records in more than one call.

 **Warning:** You can't create records for multiple object types in one call if one of those types is related to a feature in the Setup area in Salesforce. The only exceptions are the following objects:

- Custom settings objects, which are similar to custom objects. For more information, see "Custom Settings" in the Salesforce online help.

- GroupMember
- Group
- User if the UserRoleID field is not being set.

create() and Foreign Keys

You can use external ID fields as a foreign key, which allows you to create a record and relate it to another existing record in a single step instead of querying the parent record ID first. To do this, set the foreign key field to an instance of the parent sObject that only has the external ID field specified. This external ID should match the external ID value on the parent record.

The following Java and C# examples show you how to create an opportunity and relate it to an existing account using a custom external ID field named `MyExtId__c`. Each example creates an opportunity, sets the required fields, and then sets the opportunity external ID field to the account object that has only the external ID field specified. The code then creates the opportunity. Once the opportunity is created, the account will be its parent.

Java Example

```
public void createForeignKeySample() {
    try {
        Opportunity newOpportunity = new Opportunity();
        newOpportunity.setName("OpportunityWithFK");
        newOpportunity.setStageName("Prospecting");
        Calendar dt = connection.getServerTimestamp().getTimestamp();
        dt.add(Calendar.DAY_OF_MONTH, 7);
        newOpportunity.setCloseDate(dt);

        Account parentAccountRef = new Account();
        parentAccountRef.setMyExtId__c("SAP1111111");
        newOpportunity.setAccount(parentAccountRef);

        SaveResult[] results = connection
            .create(new SObject[] { newOpportunity });
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

C# Example

```
public void createForeignKeySample()
{
    try
    {
        Opportunity newOpportunity = new Opportunity();
        newOpportunity.Name = "OpportunityWithFK";
        newOpportunity.StageName = "Prospecting";
        DateTime dt = (DateTime)binding.getServerTimestamp().timestamp;
        newOpportunity.CloseDate = dt.AddDays(7);
        newOpportunity.CloseDateSpecified = true;

        // Create the parent reference.
        // Used only for foreign key reference
        // and doesn't contain any other fields
        Account accountReference = new Account();
```

```

accountReference.MyExtId__c = "SAP1111111";
newOpportunity.Account = accountReference;

// Create the account and the opportunity
SaveResult[] results = binding.create(new sObject[] {
    newOpportunity });
}

catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Creating Parent and Child Records in a Single Call Using Foreign Keys

You can use external ID fields as foreign keys to create parent and child records of different sObject types in a single call instead of creating the parent record first, querying its ID, and then creating the child record. To do this:

- Create the child sObject and populate its required fields, and optionally other fields.
- Create the parent reference sObject used only for setting the parent foreign key reference on the child sObject. This sObject has only the external ID field defined and no other fields set.
- Set the foreign key field of the child sObject to the parent reference sObject you just created.
- Create another parent sObject to be passed to the `create()` call. This sObject must have the required fields (and optionally other fields) set in addition to the external ID field.
- Call `create()` by passing it an array of sObjects to create. The parent sObject must precede the child sObject in the array, that is, the array index of the parent must be lower than the child's index.

The parent and child records are records related through a predefined relationship, such as a master-detail or lookup relationship. You can create related records that are up to 10 levels deep. Also, the related records created in a single call must have different sObject types. For more information, see [Creating Records for Different Object Types](#).

The following Java and C# examples show you how to create an opportunity with a parent account in the same `create()` call. Each example creates an Opportunity sObject and populates some of its fields, then creates two Account objects. The first account is only for the foreign key relationship, and the second is for the account creation and has the account fields set. Both accounts have the external ID field, `MyExtID__c`, set. Next, the sample calls `create()` by passing it an array of sObjects. The first element in the array is the parent sObject and the second is the opportunity sObject. The `create()` call creates the opportunity with its parent account in a single call. Finally, the sample checks the results of the call and writes the IDs of the created records to the console, or the first error if record creation fails.

Java Example

```

public void createForeignKeySample() {
    try {
        Opportunity newOpportunity = new Opportunity();
        newOpportunity.setName("OpportunityWithAccountInsert");
        newOpportunity.setStageName("Prospecting");
        Calendar dt = connection.getServerTimestamp().getTimestamp();
        dt.add(Calendar.DAY_OF_MONTH, 7);
        newOpportunity.setCloseDate(dt);

        // Create the parent reference.
        // Used only for foreign key reference
    }
}

```

```

// and doesn't contain any other fields.
Account accountReference = new Account();
accountReference.setMyExtID__c("SAP111111");
newOpportunity.setAccount(accountReference);

// Create the Account object to insert.
// Same as above but has Name field.
// Used for the create call.
Account parentAccount = new Account();
parentAccount.setName("Hallie");
parentAccount.setMyExtID__c("SAP111111");

// Create the account and the opportunity.
SaveResult[] results = connection.create(new SObject[] {
    parentAccount, newOpportunity });

// Check results.
for (int i = 0; i < results.length; i++) {
    if (results[i].isSuccess()) {
        System.out.println("Successfully created ID: "
            + results[i].getId());
    } else {
        System.out.println("Error: could not create sobject "
            + "for array element " + i + ".");
        System.out.println("The error reported was: "
            + results[i].getErrors()[0].getMessage() + "\n");
    }
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

C# Example

```

public void createForeignKeySample()
{
    try
    {
        Opportunity newOpportunity = new Opportunity();
        newOpportunity.Name = "OpportunityWithAccountInsert";
        newOpportunity.StageName = "Prospecting";
        DateTime dt = (DateTime)binding.getServerTimestamp().timestamp;
        newOpportunity.CloseDate = dt.AddDays(7);
        newOpportunity.CloseDateSpecified = true;

        // Create the parent reference.
        // Used only for foreign key reference
        // and doesn't contain any other fields.
        Account accountReference = new Account();
        accountReference.MyExtID__c = "SAP111111";
        newOpportunity.Account = accountReference;

        // Create the Account object to insert.
        // Same as above but has Name field.
    }
}

```

```

// Used for the create call.
Account parentAccount = new Account();
parentAccount.Name = "Hallie";
parentAccount.MyExtID__c = "SAP111111";

// Create the account and the opportunity.
SaveResult[] results = binding.create(new sObject[] {
    parentAccount, newOpportunity });

// Check results.
for (int i = 0; i < results.Length; i++)
{
    if (results[i].success)
    {
        Console.WriteLine("Successfully created ID: "
            + results[i].id);
    }
    else
    {
        Console.WriteLine("Error: could not create sobject "
            + "for array element " + i + ".");
        Console.WriteLine("    The error reported was: "
            + results[i].errors[0].message + "\n");
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Basic Steps for Creating Records

Creating records involves the following basic steps:

1. Create an `sObject` for one or more objects. For each record, populate its fields with the data that you want to add.
2. Construct an `sObject[]` array and populate that array with the objects that you want to create.
3. Call `create()`, passing in the `sObject[]` array.
4. Process the results in the `SaveResult[]` object to verify whether the records have been successfully created.

Sample Code—Java

This sample shows how to create records. It creates two Account objects and sets their fields. The Name of the second account isn't set so that an error occurs on creation, since Name is a required field. After making the `create()` call by passing the array containing the two accounts, the sample iterates over the results and writes the ID of the new account or an error message if the account creation fails. Finally, the sample returns an array of the new account IDs, which in this case contains only one ID.

```

public String[] createRecords() {
    // Create two accounts
}

```

```

String[] result = new String[2];
Account account1 = new Account();
Account account2 = new Account();

// Set some fields on the account object
account1.setName("The Brick Hut");
account1.setBillingStreet("403 McAdoo St");
account1.setBillingCity("Truth or Consequences");
account1.setBillingState("NM");
account1.setBillingPostalCode("87901");
account1.setBillingCountry("US");
// Required Name field is not being set on account2,
// so this record should fail during create.
// account2.setName("Camp One Creations");
account2.setBillingStreet("25800 Arnold Dr");
account2.setBillingCity("Sonoma");
account2.setBillingState("CA");
account2.setBillingPostalCode("95476");
account2.setBillingCountry("US");
Account[] accounts = { account1, account2 };

try {
    // Call create() to add the accounts
    SaveResult[] saveResults = connection.create(accounts);
    // Iterate through the results.
    // There should be one successful creation
    // and one failed creation.
    for (int i = 0; i < saveResults.length; i++) {
        if (saveResults[i].isSuccess()) {
            System.out.println("Successfully created Account ID: "
                + saveResults[i].getId());
            result[i] = saveResults[i].getId();
        } else {
            System.out.println("Error: could not create Account "
                + "for array element " + i + ".");
            System.out.println("    The error reported was: "
                + saveResults[i].getErrors()[0].getMessage() + "\n");
            result[i] = saveResults[i].getId();
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
return result;
}

```

Sample Code—C#

This sample shows how to create records. It creates two Account objects and sets their fields. The Name of the second account isn't set so that an error occurs on creation, since Name is a required field. After making the `create()` call by passing the array containing

the two accounts, the sample iterates over the results and writes the ID of the new account or an error message if the account creation fails. Finally, the sample returns an array of the new account IDs, which in this case contains only one ID.

```
public String[] createRecords()
{
    // Create two accounts
    String[] result = new String[2];
    Account account1 = new Account();
    Account account2 = new Account();

    // Set some fields on the account object
    account1.Name = "The Brick Hut";
    account1.BillingStreet = "403 McAdoo St";
    account1.BillingCity = "Truth or Consequences";
    account1.BillingState = "NM";
    account1.BillingPostalCode = "87901";
    account1.BillingCountry = "US";
    // Required Name field is not being set on account2,
    // so this record should fail during create.
    // account2.Name = "Camp One Creations";
    account2.BillingStreet = "25800 Arnold Dr";
    account2.BillingCity = "Sonoma";
    account2.BillingState = "CA";
    account2.BillingPostalCode = "95476";
    account2.BillingCountry = "US";
    Account[] accounts = { account1, account2 };

    try
    {
        // Call create() to add the accounts
        SaveResult[] saveResults = binding.create(accounts);
        // Iterate through the results.
        // There should be one successful creation
        // and one failed creation.
        for (int i = 0; i < saveResults.Length; i++)
        {
            if (saveResults[i].success)
            {
                Console.WriteLine("Successfully created Account ID: " +
                    saveResults[i].id);
                result[i] = saveResults[i].id;
            }
            else
            {
                Console.WriteLine("Error: could not create Account " +
                    "for array element " + i + ".");
            }
            Console.WriteLine("    The error reported was: " +
                saveResults[i].errors[0].message + "\n");
            result[i] = saveResults[i].id;
        }
    }
    catch (SoapException e)
    {
```

```

        Console.WriteLine("An unexpected error has occurred: " +
                           e.Message + "\n" + e.StackTrace);
    }

    return result;
}

```

Arguments

Name	Type	Description
sObjects	sObject[]	Array of one or more sObject objects to create() . Limit: 200 sObject values.

Response

[SaveResult\[\]](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[upsert\(\)](#)

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

SaveResult

The [create\(\)](#) call returns an array of SaveResult objects. Each element in the `SaveResult` array corresponds to the `sObject[]` array passed as the `sObjects` parameter in the [create\(\)](#) call. For example, the object returned in the first index in the `SaveResult` array matches the object specified in the first index of the `sObject[]` array. A `SaveResult` object has the following properties:

Name	Type	Description
id	ID	ID of the <code>sObject</code> that you attempted to create() . If this field contains a value, then the object was created successfully. If this field is empty, then the object was not created and the API returned error information instead.
success	boolean	Indicates whether the create() call succeeded (<code>true</code>) or not (<code>false</code>) for this object.
errors	Error[]	If an error occurred during the create() call, an array of one or more <code>Error</code> objects providing the error code and description. If your organization has active duplicate rules and a duplicate is detected, the <code>SaveResult</code> includes an <code>Error</code> with a data type of <code>DuplicateError</code> .

delete()

Deletes one or more records from your organization's data.

Syntax

```
DeleteResult[] = connection.delete(ID[] ids);
```

Usage

Use `delete()` to delete one or more existing records, such as individual accounts or contacts, in your organization's data. The `delete()` call is analogous to the `DELETE` statement in SQL.

Rules and Guidelines

When deleting objects, consider the following rules and guidelines:

- Your client application must be logged in with sufficient access rights to delete individual objects within the specified object. For more information, see [Factors that Affect Data Access](#).
- In addition, you might also need permission to access this object's parent object. For special access requirements, see the object's description in [Standard Objects](#).
- To ensure referential integrity, the `delete()` call supports cascading deletions. If you delete a parent object, you delete its children automatically, as long as each child object can be deleted. For example, if you delete a [Case](#), the API automatically deletes any [CaseComment](#), [CaseHistory](#), and [CaseSolution](#) objects associated with that case. However, if a [CaseComment](#) is not deletable or is currently being used, then the `delete()` call on the parent [Case](#) will fail.
- Certain objects cannot be deleted via the API. To delete an object via the `delete()` call, its object must be configured as deletable (`deletable` is `true`). To determine whether a given object can be deleted, your client application can invoke the `describeSObjects()` call on the object and inspect its `deletable` property.
- You can't delete records for multiple object types in one call if one of those types is related to a feature in the Setup area in Salesforce. The only exceptions are the following objects:
 - Custom settings objects, which are similar to custom objects. For more information, see "Custom Settings" in the Salesforce online help.
 - GroupMember
 - Group
 - User

Rollback on Error

The [AllOrNoneHeader](#) header allows you to roll back all changes unless all records are processed successfully. This header is available in API version 20.0 and later. Allows a call to roll back all changes unless all records are processed successfully.

Basic Steps for Deleting Records

Deleting records involves the following basic steps:

1. Determine the ID of each record that you want to delete. For example, you might call [query\(\)](#) to retrieve a set of records that you want to delete based on specific criteria.
2. Construct an `ID[]` array and populate it with the IDs of each record that you want to delete. You can specify the IDs of different types of objects in the same call. For example, you could specify the ID for an individual [Account](#) and an individual [Contact](#) in the same array. For information on IDs, see [ID Field Type](#).
3. Call `delete()`, passing in the `ID[]` array.
4. Process the results in the `DeleteResult[]` to verify whether the records have been successfully deleted.

Sample Code—Java

This sample shows how to delete records based on record IDs. The method in this sample accepts an array of IDs, which it passes to the `delete()` call and makes the call. It then parses the results and writes the IDs of the deleted records to the console or the first returned error if the deletion failed.

```
public void deleteRecords(String[] ids) {
    try {
        DeleteResult[] deleteResults = connection.delete(ids);
        for (int i = 0; i < deleteResults.length; i++) {
            DeleteResult deleteResult = deleteResults[i];
            if (deleteResult.isSuccess()) {
                System.out
                    .println("Deleted Record ID: " + deleteResult.getId());
            } else {
                // Handle the errors.
                // We just print the first error out for sample purposes.
                Error[] errors = deleteResult.getErrors();
                if (errors.length > 0) {
                    System.out.println("Error: could not delete " + "Record ID "
                        + deleteResult.getId() + ".");
                    System.out.println("The error reported was: (" +
                        + errors[0].getStatusCode() + ") "
                        + errors[0].getMessage() + "\n");
                }
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to delete records based on record IDs. The method in this sample accepts an array of IDs, which it passes to the `delete()` call and makes the call. It then parses the results and writes the IDs of the deleted records to the console or the first returned error if the deletion failed.

```
public void deleteRecords(String[] ids)
{
    try
    {
        DeleteResult[] deleteResults = binding.delete(ids);
```

```

for (int i = 0; i < deleteResults.Length; i++)
{
    DeleteResult deleteResult = deleteResults[i];
    if (deleteResult.success)
    {
        Console.WriteLine("Deleted Record ID: " + deleteResult.id);
    }
    else
    {
        // Handle the errors.
        // We just print the first error out for sample purposes.
        Error[] errors = deleteResult.errors;
        if (errors.Length > 0)
        {
            Console.WriteLine("Error: could not delete " + "Record ID "
                + deleteResult.id + ".");
            Console.WriteLine("    The error reported was: (" +
                + errors[0].statusCode + ") "
                + errors[0].message + "\n");
        }
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}

```

Arguments

Name	Type	Description
ids	ID[]	Array of one or more IDs associated with the objects to delete. In version 7.0 and later, you can pass a maximum of 200 object IDs to the delete () call. In version 6.0 and earlier, the limit is 2,000.

Response

[DeleteResult\[\]](#)

Faults

[InvalidSObjectFault](#)

UnexpectedErrorFault

SEE ALSO:

[API Call Basics](#)https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

DeleteResult

The `delete()` call returns an array of `DeleteResult` objects. Each element in the `DeleteResult` array corresponds to the `ID[]` array passed as the `ids` parameter in the `delete()` call. For example, the object returned in the first index in the `DeleteResult` array matches the object specified in the first index of the `ID[]` array.

A `DeleteResult` object has the following properties:

Name	Type	Description
<code>id</code>	<code>ID</code>	ID of an <code>sObject</code> that you attempted to delete. For information on IDs, see ID Field Type .
<code>success</code>	<code>boolean</code>	Indicates whether the <code>delete()</code> call succeeded (<code>true</code>) or not (<code>false</code>) for this object.
<code>errors</code>	<code>Error[]</code>	If an error occurred during the <code>delete()</code> call, an array of one or more <code>Error</code> objects providing the error information.

emptyRecycleBin()

Delete records from the recycle bin immediately.

Syntax

```
EmptyRecycleBinResult[] = connection.emptyRecycleBin(ID[] ids);
```

Usage

The Recycle Bin lets you view and restore recently deleted records for 15 days before they are permanently deleted. Your org can have up to 5,000 records per license in the Recycle Bin at any one time. For example, if your org has five user licenses, 25,000 records can be stored in the Recycle Bin. If your org reaches its Recycle Bin limit, Salesforce automatically removes the oldest records, as long as they have been in the recycle bin for at least two hours.

If you know you will be adding a great number of records to the Recycle Bin and you know you won't need to `undelete()` them, you may wish to remove them before the Salesforce process deletes records. For example, you can use this call if you are loading a large number of records for testing, or if you are doing a large number of `create() calls` followed by `delete() calls`.

Rules and Guidelines

When emptying recycle bins, consider the following rules and guidelines:

- The logged in user can delete any record that he or she can query in their Recycle Bin, or the recycle bins of any subordinates. If the logged in user has Modify All Data permission, he or she can query and delete records from any Recycle Bin in the organization.

- Available in version 10.0 and later.
- Maximum number of records is 200.
- Do not include the IDs of any records that will be cascade deleted, or an error will occur.
- Once records are deleted using this call, they cannot be [undelete\(\)d](#).
- After records are deleted from the Recycle Bin using this call, they can be queried using [queryAll\(\)](#) for some time. Typically this time is 24 hours, but may be shorter or longer.

Sample Code—Java

This sample shows how to empty the Recycle Bin. It accepts an array containing the IDs of the records to remove from the Recycle Bin. It calls `emptyRecycleBin()` and passes it the array of IDs. Next, it iterates over the results and writes the IDs of the removed records or the first error of the failed records to the console.

```
public void emptyRecycleBin(String[] ids) {
    try {
        EmptyRecycleBinResult[] emptyRecycleBinResults = connection
            .emptyRecycleBin(ids);
        for (int i = 0; i < emptyRecycleBinResults.length; i++) {
            EmptyRecycleBinResult emptyRecycleBinResult = emptyRecycleBinResults[i];
            if (emptyRecycleBinResult.isSuccess()) {
                System.out.println("Recycled ID: "
                    + emptyRecycleBinResult.getId());
            } else {
                Error[] errors = emptyRecycleBinResult.getErrors();
                if (errors.length > 0) {
                    System.out
                        .println("Error code: " + errors[0].getStatusCode());
                    System.out
                        .println("Error message: " + errors[0].getMessage());
                }
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to empty the Recycle Bin. It accepts an array containing the IDs of the records to remove from the Recycle Bin. It calls `emptyRecycleBin()` and passes it the array of IDs. Next, it iterates over the results and writes the IDs of the removed records or the first error of the failed records to the console.

```
public void emptyRecycleBin(String[] ids)
{
    try
    {
        EmptyRecycleBinResult[] emptyRecycleBinResults =
            binding.emptyRecycleBin(ids);
        for (int i = 0; i < emptyRecycleBinResults.Length; i++)
        {
```

```

EmptyRecycleBinResult emptyRecycleBinResult = emptyRecycleBinResults[i];
if (emptyRecycleBinResult.success)
{
    Console.WriteLine("Recycled ID: "
        + emptyRecycleBinResult.id);
}
else
{
    Error[] errors = emptyRecycleBinResult.errors;
    if (errors.Length > 0)
    {
        Console.WriteLine("Error code: " + errors[0].statusCode);
        Console.WriteLine("Error message: " + errors[0].message);
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Arguments

Name	Type	Description
ids	ID[]	Array of one or more IDs associated with the records to delete from the Recycle Bin. Maximum number of records is 200.

Response

[EmptyRecycleBinResult](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[delete\(\)](#)

[undelete\(\)](#)

EmptyRecycleBinResult

The `emptyRecycleBin()` call returns an array of `EmptyRecycleBinResult` objects. Each element in the array corresponds to an element in the `ID[]` array passed as the parameter in the `emptyRecycleBin()` call. For example, the object returned in the first index in the `EmptyRecycleBinResult` array matches the object specified in the first index of the `ID[]` array.

A `EmptyRecycleBinResult` object has the following properties:

Name	Type	Description
<code>id</code>	<code>ID</code>	ID of an <code>sObject</code> that you attempted to delete from the Recycle Bin. For information about IDs, see ID Field Type .
<code>isSuccess</code>	<code>boolean</code>	Indicates whether the call succeeded (<code>true</code>) or not (<code>false</code>) for this record.
<code>errors</code>	<code>Error[]</code>	If an error occurred during the call, an array of one or more <code>Error</code> objects providing the error information.

executeListView()

Executes a list view's SOQL query to retrieve data, labels, and actions from a list view.

Syntax

```
ExecuteListViewResult result = connection.executeListView(ExecuteListViewResult request);
```

Usage

The `executeListView()` call takes an `ExecuteListViewRequest` object, executes the SOQL query for the list view, and returns the resulting data and presentation information in an `ExecuteListViewResult` object. This call is available in API version 32.0 and later.

Sample Code—Java

```
private void example(ApiProtocol protocol, AppVersion version) throws Exception {

    // Get the list results via the list view API
    EnterpriseConnection connection =
makeClient(getUserUtil().getUserWithModifyAllData(), AppVersion.VERSION_190,
        getName());
    ExecuteListViewRequest request = new ExecuteListViewRequest();
    request.setSObjectType("Account");
    request.setDeveloperNameOrId(listViews[0].getId());
    request.setLimit(50000);

    com.sforce.soap.enterprise.ExecuteListViewResult result =
connection.executeListView(request);
}
```

Arguments

Name	Type	Description
request	ExecuteListViewRequest	An object that specifies the list view and the limit, offset, and ordering of the results.

Response

An [ExecuteListViewResult](#) object.

ExecuteListViewRequest

Use the `ExecuteListViewRequest` object with `executeListView()` to retrieve data, labels, and actions from a list view.

The `ExecuteListViewRequest` object has the following properties:

Name	Type	Description
developerNameOrId	string	The list view's ID or fully qualified developer name.
limit	int	The maximum number of records to return. Default: 25
offset	int	The number of records to skip. Default: 0
orderBy	ListViewOrderBy[]	The order in which to return the records.
sObjectType	string	The API name of the sObject for the list view.

ExecuteListViewResult

Contains list view data that you retrieve programmatically.

To retrieve an `executeListViewResult` object, use the `executeListView()` call. The `executeListViewResult` object has the following properties:

Name	Type	Description
columns	ListViewColumn[]	An array of the columns in the list view.
developerName	string	The list view's fully qualified developer name.
done	boolean	If <code>true</code> , indicates that all records have been returned.
id	ID	The list view's ID.
label	string	The display label of the list view.
records	ListViewRecord[]	An array of records that match the list view query.
size	int	The number of records that are returned by the list view query.

ListViewColumn

Contains metadata about a single list view column.

The `ListViewColumn` object is returned by the `describeSoqlListViews()` and `executeListView()` calls. It has the following properties:

Name	Type	Description
<code>ascendingLabel</code>	string	The localized type-specific label for sorting the column in ascending order. For example: "A-Z" for a text field, or "Low to High" for a numeric field. Set to null if the column isn't sortable.
<code>descendingLabel</code>	string	The localized type-specific label for sorting the column in descending order. For example: "Z-A" for a text field, or "High to Low" for a numeric field. Set to null if the column is not sortable.
<code>fieldNameOrPath</code>	string	The field name or SOQL field path for the column.
<code>hidden</code>	boolean	If true, specifies that the column is not displayed, and is present only to support the display of other columns or other client-side logic.
<code>label</code>	string	The localized display label for the column.
<code>selectListItem</code>	string	The SOQL SELECT item for the column. The item might differ from the field name or path, due to display formatting (for example, <code>toLabel</code> for picklists).
<code>sortDirection</code>	<code>orderByDirection</code>	An enumerated value, one of the following if the column is sortable: <ul style="list-style-type: none"> • ascending • descending Set to null if the column is not sortable.
<code>sortIndex</code>	int	The zero-based index that indicates the column's position within a multilevel sort, or null if the records are not sorted by the column.
<code>sortable</code>	boolean	Whether the column is sortable, in which case it might be referenced in the <code>ExecuteListView</code> <code>orderBy</code> parameter.
<code>type</code>	<code>FieldType</code>	The column data type.

ListViewRecord

Represents a single row in a list view.

The `ListViewRecord` object is a member of the `ExecuteListViewResult` object and has the following properties:

Name	Type	Description
<code>columns</code>	<code>ListViewRecordColumn[]</code>	The columns and their values for the record. The record data columns are returned in the same order as metadata and describe columns. For any data column that's obtained by using <code>ExecuteListViewResult.getRecords() [0].getColumns [index]</code> ,

Name	Type	Description
		the corresponding describe column can be obtained with <code>ExecuteListViewResult.getColumns[index]</code> .

ListViewRecordColumn

Represents a single cell in a row from a list view.

The `ListViewRecordColumn` object is one cell (column) of a row ([ListViewRecord](#)) and has the following properties:

Name	Type	Description
<code>fieldNameOrPath</code>	string	The field name or SOQL field path for the column.
<code>value</code>	string	The contents of the record for a certain column, localized if appropriate, or null if there's no value.

findDuplicates ()

Performs rule-based searches for duplicate records. The input is an array of `sObject`, each of which specifies the values to search for and the type of object that supplies the duplicate rules. The output identifies the detected duplicates for each object that supplies the duplicate rules. `findDuplicates ()` applies the rules to the values to do the search. The output identifies the detected duplicates for each `sObject`.

Syntax

```
FindDuplicatesResult[] duplicateResults =
    connection.findDuplicates(sObject[] inputSObjectArray);
```

Usage

Use `findDuplicates ()` to apply duplicate rules associated with an object to values specified by each `sObject`. Each `sObject` also has a type that corresponds to an object.

`findDuplicates ()` uses the duplicate rules for the object that has the same type as the `sObject`. For example, if the `sObject` type is `Account`, `findDuplicates ()` uses the duplicate rules associated with the `Account` object.



Note:

- All the `sObject` elements in the input array must have the same type, and that type must correspond to an object type that supports duplicate rules.
- The input array is limited to 50 elements. If you exceed this limit, the SOAP call returns an [API Fault Element](#) containing the following fields:
 - `ExceptionCode: LIMIT_EXCEEDED`
 - `exceptionMessage: Configuration error: The number of records to check is greater than the permitted batch size.`

For each input `sObject`, `findDuplicates()` adds a `FindDuplicatesResult` object to the output array.

Matching is controlled by the values specified in the `sObject`. The values can include a record ID, a field map, or both. The specified values determine the behavior of `findDuplicates()`:

Record ID only

`findDuplicates()` searches the object defined by the duplicate rule for an existing record that has the same ID. Then it loads the values from that record, and searches for duplicates based on those values.

Field Map only

`findDuplicates()` loads the values from the map and searches for duplicates based on those values.

Record ID and Field Map

`findDuplicates()` searches the object defined by the duplicate rule for an existing record that has the same ID. It loads any values from that record that aren't specified in the map, and then loads values from the map. Based on the resulting union of values, `findDuplicates()` searches for duplicates.

The output of `findDuplicates()` is an array of `FindDuplicatesResult` objects with the same number of elements as the input array, and in the same order. The output objects encapsulate record IDs for duplicate records, if any. Optionally, the output objects also contain values from the duplicate records.

Each `FindDuplicatesResult` element contains a `DuplicateResult` object. If `findDuplicates()` doesn't find any duplicates for an `sObject`, the `duplicateRule` field in `DuplicateResult` contains the name of the duplicate rule that `findDuplicates()` applied, but the `matchResults` array is empty.

If the `includeRecordDetails` flag in `DuplicateRuleHeader` is set to `false`, `findDuplicates()` only returns the record IDs of the matching records. Otherwise, `findDuplicates()` returns all the fields specified in the primary `CompactLayout` associated with the target object.

Basic Steps for Using

1. Create one or more `sObject` objects with a type that corresponds to the object that has the duplicate rules you want to use.
2. In each `sObject`, specify record IDs or field maps (or both) to compare to records in the object.
3. Set `DuplicateRuleHeader` to control the output you want.

Sample

The following Java sample demonstrates how to search for duplicates of a Lead, using the standard Leads duplicate rule.

```
SObject[] inputSObjectArray = new SObject[1];
// Instantiate an empty Java SObject
SObject searchCriteria = new SObject();
// Set its type to Lead. This tells findDuplicates() to use the duplicate rules
// for Lead
searchCriteria.setType("Lead");
/*
 * Set the necessary fields for matching, based on the standard matching rules
 * for Lead (Search help.salesforce.com for "Standard Contact and Lead Matching
 * Rule" to see the rules).
 */
searchCriteria.setField("FirstName", "Mark");
searchCriteria.setField("LastName", "Benioff");
searchCriteria.setField("Company", "Salesforce.com Inc");
searchCriteria.setField("Title", "CEO");
```

```
searchCriteria.setField("Email", "ceo@salesforce.com");
// Add the sObject to the input array
inputSObjectArray[0] = searchCriteria;
/*
 * By default, findDuplicates() returns only record IDs. To return
 * additional values, set the second parameter to true.
 */
connection.setDuplicateRuleHeader( /* @param allowSave - Not Applicable
                                    for this API call */ false,
                                    /* @param includeRecordDetails */ false,
                                    /* @param runAsCurrentUser - Not
                                       Applicable for this API call */ false);

// Invoke findDuplicates() to find duplicates based on the information in the
// SObject array
FindDuplicatesResult[] callResults =
    connection.findDuplicates(inputSObjectArray);

if (callResults == null || callResults.length == 0) {
    System.out.println("Major error while calling findDuplicates(). Cannot proceed");
    return;
}

// Iterate through the results
// For each SObject in the input array, get the duplicate results
for (FindDuplicatesResult findDupResult : callResults) {
    // If errors were found for this SObject, print them out
    if (!findDupResult.isSuccess()) {
        for (Error findDupError : findDupResult.getErrors()) {
            System.out.println("FindDuplicatesRule errors detected: " +
                               findDupError.getMessage());
        }
    } else {
        /*
         * Get the DuplicateResult object array for the result. Each element in
         * the array represents the result of testing one duplicate rule for the
         * SObject. Process each DuplicateResult.
         * SObject. Process each DuplicateResult
         */
        for (DuplicateResult dupeResult : findDupResult.getDuplicateResults()) {
            System.out.println("Duplicate rule: " +
                               dupeResult.getDuplicateRule());
            // Print out the name of the object associated with the duplicate
            // rule
            System.out.println("Source of this duplicate rule is: " +
                               dupeResult.getDuplicateRuleEntityType());
            for (MatchResult matchResult : dupeResult.getMatchResults()) {
                if (!matchResult.isSuccess()) {
                    for (Error e : matchResult.getErrors()) {
                        System.out.println("Errors detected: " +
                                           e.getMessage());
                    }
                } else {
                    System.out.println("Matching rule is: " +
```

```
        matchResult.getRule());
    System.out.println("Object type for this matching rule is: " +
        matchResult.getEntityType());
    for (MatchRecord matchRecord : matchResult.getMatchRecords()) {
        System.out.println("Duplicate record ID: " +
            matchRecord.getRecord().getId());
    }
}
}
}
}
```

Arguments

Name	Type	Description
sObjects	Array of sObject	Required. A list of sObject objects that contain values you want to search for.

Response

An array of `FindDuplicatesResult` objects.

FindDuplicatesResult

Represents the result of a duplicate search for a single sObject in the input array. Because the object associated with the sObject can have more than one duplicate rule, `FindDuplicatesResult` contains an array of `DuplicateResult` objects.

Fields

Field Name	Field Type	Description
duplicateResults	Array of DuplicateResult objects	The result of each duplicate rule applied by <code>findDuplicates()</code> to a single sObject.
errors	Array of Error objects	Contains an array of errors encountered by <code>findDuplicates()</code> . The elements don't correspond to elements in the input array. For example, if the input array <code>sObjects</code> contains 10 entries, but <code>findDuplicates()</code> encounters only one error, <code>errors</code> contains one element.
success	boolean	This field is set to <code>true</code> if the <code>findDuplicates()</code> doesn't encounter any errors.

Faults

[InvalidSObjectFault](#)
[UnexpectedErrorFault](#)
[InvalidFieldFault](#)

getDeleted()

Retrieves the list of individual records that have been deleted within the given timespan for the specified object.

Syntax

```
GetDeletedResult = connection.getDeleted(string sObjectType, dateTime startDate, dateTime  
EndDate);
```

Usage

Use [getDeleted\(\)](#) for data replication applications to retrieve a list of records that have been deleted from your organization's data within the specified timespan. The [getDeleted\(\)](#) call retrieves a [GetDeletedResult](#) object that contains an array of [DeletedRecord](#) objects containing the ID of each deleted record and the date/time (Coordinated Universal Time (UTC) time zone) on which it was deleted. Be sure to read [Data Replication](#) before using [getDeleted\(\)](#) in your client applications. (For information on IDs, see [ID Field Type](#).)

As of release 8.0, the [getDeleted\(\)](#) call respects the user's sharing model.

Rules and Guidelines

When replicating deleted records, consider the following rules and guidelines:

- The specified `startDate` must chronologically precede the specified `endDate` value. The specified `startDate` cannot be the same value as, or later than, the specified `endDate` value. Otherwise, the API returns an `INVALID_REPLICATION_DATE` error.
- Records are returned only if the user has access to them.
- Results are returned for no more than 15 days previous to the day the call is executed (or earlier if an administrator has purged the Recycle Bin). If the purge has been performed before your [getDeleted\(\)](#) call is executed, an `INVALID_REPLICATION_DATE` error is returned.
- If `latestDateCovered` is less than `endDate`, the call will fail, returning an `INVALID_REPLICATION_DATE` error with the value of `latestDateCovered`.
- Deleted records are written to a delete log, which [getDeleted\(\)](#) accesses. A background process that runs every two hours purges records that have been in an organization's delete log for more than two hours if the number of records is above a certain limit. Starting with the oldest records, the process purges delete log entries until the delete log is back below the limit. This is done to protect Salesforce from performance issues related to massive delete logs. The limit is calculated using this formula:

```
5000 * number of licenses in the organization
```

For example, an organization with 1,000 licenses could have up to 5,000,000 (five million) records in the delete log before any purging took place. If purging has been performed before your [getDeleted\(\)](#) call is executed, an `INVALID_REPLICATION_DATE` error is returned. If you get this exception, you should do a full pull of the table.

- If you delete a large numbers of records, your data replication should run more frequently than every two hours to ensure all records are returned by `getDeleted()`.
- Client applications typically poll for changed data periodically. For important polling considerations, see [Polling for Changes](#).
- Records for certain objects cannot be replicated via the API. To replicate a record via the `getDeleted()` call, its object must be configured as replicable (`replicable` is `true`). To determine whether a given object can be replicated, your client application can invoke the `describeSObjects()` call on the object and inspect its `replicable` property.
- Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time. To determine how your development tool handles time values, refer to its documentation.
- If you call `getDeleted()` for a history object, the call returns the records deleted during the given date range for all history objects, not only the history object you specified. For example, if you call `getDeleted()` for AccountHistory, you'll get records deleted during the given date range for AccountHistory, ContactHistory, and so on.

Basic Steps for Replicating Deleted Records

You can replicate deleted records using the following basic steps for each object:

1. Optionally, determine whether the structure of the object has changed since the last replication request, as described in [Checking for Structural Changes in the Object](#).
2. Call `getDeleted()`, passing in the object and the relevant time span for deleted records.
3. In the `DeleteResult` object, iterate through the returned array of `DeletedRecord` objects containing the ID of each deleted record and the date on which it was deleted (Coordinated Universal Time (UTC) time zone).
4. Take the appropriate action on the local data to remove the deleted records or flag as deleted.
5. Optionally, save the request time span for future reference. You should save the value of `latestDateCovered`.

A client application likely performs other tasks associated with data replication operations. For example, if an opportunity is closed, a client application might run a new revenue report. Similarly, if a task is completed, the process might log this in another system.

Sample Code—Java

This sample calls `getDeleted()` to get all accounts that were deleted in the last 60 minutes. It then writes the ID and the deleted date of each returned account to the console.

```
public void getDeletedRecords() {  
    try {  
        GregorianCalendar endTime = (GregorianCalendar)  
            connection.getServerTimestamp().getTimestamp();  
        GregorianCalendar startTime = (GregorianCalendar) endTime.clone();  
        // Subtract 60 minutes from the server time so that we have  
        // a valid time frame.  
        startTime.add(GregorianCalendar.MINUTE, -60);  
        System.out.println("Checking deletes at or after: "  
            + startTime.getTime().toString());  
  
        // Get records deleted during the specified time frame.  
        GetDeletedResult gdResult = connection.getDeleted("Account",  
            startTime, endTime);  
  
        // Check the number of records contained in the results,  
    } catch (Exception e) {  
        System.out.println("An error occurred: " + e.getMessage());  
    }  
}
```

```

// to check if something was deleted in the 60 minute span.
DeletedRecord[] deletedRecords = gdResult.getDeletedRecords();
if (deletedRecords != null && deletedRecords.length > 0) {
    for (int i = 0; i < deletedRecords.length; i++) {
        DeletedRecord dr = deletedRecords[i];
        System.out.println(dr.getId() + " was deleted on "
            + dr.getDeletedDate().getTime().toString());
    }
} else {
    System.out.println("No deletions of Account records in "
        + "the last 60 minutes.");
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

Sample Code—C#

This sample calls `getDeleted()` to get all accounts that were deleted in the last 60 minutes. It then writes the ID and the deleted date of each returned account to the console.

```

public void getDeletedRecords()
{
    try
    {
        DateTime endTime = binding.getServerTimestamp().timestamp;
        // Subtract 60 minutes from the server time so that we have
        // a valid time frame.
        DateTime startTime = endTime.AddMinutes(-60);
        Console.WriteLine("Checking deletes at or after: "
            + startTimeToLocalTime().ToString());

        // Get records deleted during the specified time frame.
        GetDeletedResult gdResult = binding.getDeleted("Account",
            startTime, endTime);

        // Check the number of records contained in the results,
        // to check if something was deleted in the 60 minute span.
        DeletedRecord[] deletedRecords = gdResult.deletedRecords;
        if (deletedRecords != null && deletedRecords.Length > 0)
        {
            for (int i = 0; i < deletedRecords.Length; i++)
            {
                DeletedRecord dr = deletedRecords[i];
                Console.WriteLine(dr.id + " was deleted on "
                    + dr.deletedDateToLocalTime().ToString());
            }
        }
        else
        {
            Console.WriteLine("No deletions of Account records in "
                + "the last 60 minutes.");
        }
    }
}

```

```

        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
                           e.Message + "\n" + e.StackTrace);
    }
}

```

Arguments

Name	Type	Description
sObjectTypeEntityType	string	Object type. The specified value must be a valid object for your organization. See sObject .
startDate	dateTime	Starting date/time (Coordinated Universal Time (UTC)—not local—timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:30:15 is interpreted as 12:30:00 UTC).
endDate	dateTime	Ending date/time (Coordinated Universal Time (UTC)—not local—timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:35:15 is interpreted as 12:35:00 UTC).

Limits

There are record limits on the result [GetDeletedResult](#):

- If your `getDeleted()` call returns more than 600,000 records and the user is a system administrator, an exception EXCEEDED_ID_LIMIT is returned.
- If your `getDeleted()` call returns more than 20,000 records and the user is not a system administrator, an exception OPERATION_TOO_LARGE is returned. Note that this error is returned when more than 20,000 records across the organization have been deleted, not just the records viewable by the user.

You can correct the error by choosing start and end dates that are closer together.

Response

[GetDeletedResult](#)

Faults

[InvalidSObjectFault](#)

UnexpectedErrorFault

SEE ALSO:

[Data Replication](#)

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

GetDeletedResult

The `getDeleted()` call returns a `GetDeletedResult` object that contains an array of `DeletedRecord` records and two properties:

Name	Type	Description
earliestDateAvailable	<code>dateTime</code>	For the object type of the <code>getDeleted()</code> call, the timestamp (Coordinated Universal Time (UTC)—not local—timezone) of the last physically deleted object. If this value is less than <code>endDate</code> , the call will fail, and you should resynch your data before performing another replication.
deletedRecords []	<code>deletedRecords</code>	Array of the deleted records which satisfy the start and end dates specified in the <code>getDeleted()</code> call.
latestDateCovered	<code>dateTime</code>	The timestamp (Coordinated Universal Time (UTC)—not local—time zone) of the last date covered in the <code>getDeleted()</code> call. If there is a value, it is less than or equal to <code>endDate</code> . A value here indicates that, for safety, you should use this value for the <code>startDate</code> of your next call to capture the changes that started after this date but did not complete before <code>endDate</code> and were, therefore, not returned in the previous call.

deletedRecords

The `GetDeletedResult` contains an array of `deletedRecords`, which contain the following properties:

Name	Type	Description
<code>deletedDate</code>	<code>dateTime</code>	Date and time (Coordinated Universal Time (UTC)—not local—timezone) when this record was deleted.
<code>id</code>	<code>ID</code>	ID of an <code>sObject</code> that has been deleted.

getUpdated()

Retrieves the list of individual records that have been updated (added or changed) within the given timespan for the specified object.

Syntax

```
GetUpdatedResult[] = connection.getUpdated(string sObjectType, dateTime startDate, dateTime  
endDate);
```

Usage

Use [getUpdated\(\)](#) for data replication applications to retrieve a set of IDs for objects of the specified object that have been created or updated within the specified timespan. The [getUpdated\(\)](#) call retrieves an array of [GetUpdatedResult](#) objects containing the ID of each created or updated object and the date/time (Coordinated Universal Time (UTC) time zone) on which it was created or updated, respectively. Be sure to read [Data Replication](#) before using [getUpdated\(\)](#) in your client application.



Note: The [getUpdated\(\)](#) call retrieves the IDs only for objects to which the logged-in user has access.

Rules and Guidelines

When replicating created and updated objects, consider the following rules and guidelines:

- The specified `startDate` must chronologically precede the specified `endDate` value. The specified `startDate` cannot be the same value as, or later than, the specified `endDate` value. Otherwise, the API returns an `INVALID_REPLICATION_DATE` error.
- Results are returned for no more than 30 days previous to the day the call is executed.
- Client applications typically poll for changed data periodically. For important polling considerations, see [Polling for Changes](#).
- Your client application can replicate any objects to which it has sufficient permissions. For example, to replicate all data for your organization, your client application must be logged in with "View All Data" access rights to the specified object. Similarly, the objects must be within your sharing rules. For more information, see [Factors that Affect Data Access](#).
- Certain objects cannot be replicated via the API. To replicate an object via the [getUpdated\(\)](#) call, its object must be configured as replicateable (`replicable` is `true`). To determine whether a given object can be replicated, your client application can invoke the [describeSObjects\(\)](#) call on the object and inspect its `replicable` property.
- Certain objects cannot be deleted, such as [Group](#), [User](#), [Contract](#), or [Product2](#) objects. However, if instances of these objects are no longer visible in the Salesforce user interface, they may have been rendered inactive so that only users with administrative access can see them. To determine whether a missing object instance has been made inactive, your client application can call [getUpdated\(\)](#) and check the object's active flag.
- Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time. To determine how your development tool handles time values, refer to its documentation.

Basic Steps for Replicating Updated Objects

Replicating objects involves the following basic steps for each object that you want to replicate:

1. Optionally, the client application determines whether the structure of the object has changed since the last replication request, as described in [Checking for Structural Changes in the Object](#).
2. Call [getUpdated\(\)](#), passing in the object and timespan for which to retrieve data.
3. Iterate through the returned array of IDs. For each ID element in the array, call [retrieve\(\)](#) to obtain the latest information you want from the associated object. Your client application must then take the appropriate action on the local data, such as inserting new rows or updating existing ones with the latest information.

- 4.** Optionally, the client application saves the request timestamp for future reference.

A client application likely performs other tasks associated with data replication operations. For example, if an opportunity were to become closed, a client application might run a new revenue report. Similarly, if a task were completed, the process might log this somehow in another system.

Sample Code—Java

This sample gets the accounts that were updated in the last 60 minutes and writes their IDs to the console.

```
public void getUpdatedRecords() {
    try {
        GregorianCalendar endTime = (GregorianCalendar) connection
            .getServerTimestamp().getTimestamp();
        GregorianCalendar startTime = (GregorianCalendar) endTime.clone();
        // Subtract 60 minutes from the server time so that we have
        // a valid time frame.
        startTime.add(GregorianCalendar.MINUTE, -60);
        System.out.println("Checking updates as of: "
            + startTime.getTime().toString());

        // Get the updated accounts within the specified time frame
        GetUpdatedResult ur = connection.getUpdated("Account", startTime,
            endTime);
        System.out.println("GetUpdateResult: " + ur.getIds().length);

        // Write the results
        if (ur.getIds() != null && ur.getIds().length > 0) {
            for (int i = 0; i < ur.getIds().length; i++) {
                System.out.println(ur.getIds()[i] + " was updated between "
                    + startTime.getTime().toString() + " and "
                    + endTime.getTime().toString());
            }
        } else {
            System.out.println("No updates to accounts in "
                + "the last 60 minutes.");
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample gets the accounts that were updated in the last 60 minutes and writes their IDs to the console.

```
public void getUpdatedRecords()
{
    try
    {
        DateTime endTime = binding.getServerTimestamp().timestamp;
        // Subtract 60 minutes from the server time so that we have
        // a valid time frame.
```

```

DateTime startTime = endTime.AddMinutes(-60);
Console.WriteLine("Checking updates as of: "
    + startTime.ToLocalTime().ToString());

// Get the updated accounts within the specified time frame
GetUpdatedResult ur = binding.getUpdated("Account", startTime,
    endTime);
Console.WriteLine("GetUpdateResult: " + ur.ids.Length);

// Write the results
if (ur.ids != null & ur.ids.Length > 0)
{
    for (int i = 0; i < ur.ids.Length; i++)
    {
        Console.WriteLine(ur.ids[i] + " was updated between "
            + startTime.ToLocalTime().ToString() + " and "
            + endTime.ToLocalTime().ToString());
    }
}
else
{
    Console.WriteLine("No updates to accounts in "
        + "the last 60 minutes.");
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Arguments

Name	Type	Description
sObjectTypeEntityType	string	Object type. The specified value must be a valid object for your organization. For a list of standard objects, see Standard Objects .
startDate	dateTime	Starting date/time (Coordinated Universal Time (UTC) time zone—not local—timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:30:15 is interpreted as 12:30:00 UTC).
endDate	dateTime	Ending date/time (Coordinated Universal Time (UTC) time zone—not local—timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:35:15 is interpreted as 12:35:00 UTC).

- !** **Important:** There is a limit of 600,000 IDs in the result `GetUpdatedResult[]`. If your `getUpdated()` call returns more than 600,000 IDs, an exception EXCEEDED_ID_LIMIT is returned. You can correct the error by choosing start and end dates that are closer together.

Response

[GetUpdatedResult](#)[]

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[Data Replication](#)

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

GetUpdatedResult

The `getUpdated()` call returns a `GetUpdatedResult` object that contains information about each record that was inserted or updated within the given timespan. An `GetUpdatedResult` object has the following properties:

Name	Type	Description
<code>id[]</code>	<code>ID</code>	Array of IDs of each object that has been updated.
<code>latestDateCovered</code>	<code>dateTime</code>	The timestamp (Coordinated Universal Time (UTC)—not local—time zone) of the last date covered in the <code>getUpdated()</code> call. If there is a value, it is less than or equal to <code>endDate</code> . A value here indicates that, for safety, you should use this value for the <code>startDate</code> of your next call to capture the changes that started after this date but did not complete before the <code>endDate</code> and were, therefore, not returned in the previous call.  Note: If Salesforce executes a long-running transaction on your instance, the value in this field is the start time of that long-running transaction until it completes. This is because a long-running transaction might affect your user data (for example, batch processing).

invalidateSessions()

Ends one or more sessions specified by a `sessionId`.

Syntax

```
InvalidateSessionsResult = connection.invalidateSessions(string[] sessionIds);
```

Usage

Use this call to end one or more sessions.

You can also use `logout()` to end just one session, the session of the logged-in user.

Sample Code—Java

This sample invalidates a set of sessions. The method in this sample takes an array of session IDs passed in as String values. The method then calls `invalidateSessions()` with this array and then checks the results for any errors.

```
public void invalidateSessionsSample(String[] sessionIds) {
    try {
        InvalidateSessionsResult[] results;
        results = connection.invalidateSessions(sessionIds);
        for (InvalidateSessionsResult result : results) {
            // Check results for errors
            if (!result.isSuccess()) {
                if (result.getErrors().length > 0) {
                    System.out.println("Status code: "
                        + result.getErrors()[0].getStatusCode());
                    System.out.println("Error message: "
                        + result.getErrors()[0].getMessage());
                }
            } else {
                System.out.println("Success.");
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample invalidates a set of sessions. The method in this sample takes an array of session IDs passed in as String values. The method then calls `invalidateSessions()` with this array and then checks the results for any errors.

```
public void invalidateSessionsSample(string[] sessionIds)
{
    try
    {
        InvalidateSessionsResult[] results;
        results = binding.invalidateSessions(sessionIds);
        foreach (InvalidateSessionsResult result in results)
        {
            // Check results for errors
            if (!result.success)
            {
                if (result.errors.Length > 0)
                {
                    Console.WriteLine("Status code: " +
                        result.errors[0].statusCode);
                }
            }
        }
    }
}
```

```

        Console.WriteLine("Error message: " +
            result.errors[0].message);
    }
}
else
{
    Console.WriteLine("Success.");
}
}
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Arguments

Name	Type	Description
sessionIds	string[]	One or more <code>sessionId</code> strings. Limit 200. You can obtain your <code>sessionId</code> from the <code>SessionHeader</code> .

Response

[InvalidateSessionsResult\[\]](#)

Faults

[UnexpectedErrorFault](#)

InvalidateSessionsResult

The `invalidateSessions ()` call returns an array of `LogoutResult` objects. Each object has the following properties:

Name	Type	Description
success	boolean	Indicates whether the session was successfully terminated (<code>true</code>) or not (<code>false</code>).
errors	Error[]	If an error occurred during the call, an array of one or more <code>Error</code> objects. Each object contains an error code and description.

login()

Logs in to the login server and starts a client session.

 **Note:** `login()` calls count toward your API usage limits.

Syntax

```
LoginResult = connection.login(string username, string password);
```

Usage

Use the `login()` call to log in to the login server and start a client session. A client application must log in and obtain a `sessionId` and server URL before making other API calls.

When a client application invokes the `login()` call, it passes in a username and password as user credentials. Upon invocation, the API authenticates the credentials. It then returns the `sessionId`, the user ID associated with the logged-in username, and a URL that points to the Force.com API to use in all subsequent API calls.

Salesforce checks the IP address from which the client application is logging in, and blocks logins from unknown IP addresses. For a blocked login via the API, Salesforce returns a login fault. Then, the user must add their security token to the end of their password in order to log in. A security token is an automatically-generated key from Salesforce. For example, if a user's password is `mypassword`, and their security token is `xxxxxxxxxx`, then the user must enter `mypasswordxxxxxxxxxx` to log in. Users can obtain their security token by changing their password or resetting their security token via the Salesforce user interface. When a user changes their password or resets their security token, Salesforce sends a new security token to the email address on the user's Salesforce record. The security token is valid until a user resets their security token, changes their password, or has their password reset. When the security token is invalid, the user must repeat the login process to log in. To avoid this, the administrator can make sure the client's IP address is added to the organization's list of trusted IP addresses. For more information, see [Security Token](#).

After logging in, make sure that your client application performs these tasks:

- Sets the session ID in the SOAP header so that the API can validate subsequent requests for this session.
- Specifies the server URL as the target for subsequent service requests. The login server supports only login calls.

Development tools differ in the way you specify session headers and server URLs. For more information, see the documentation for your particular development tool.

 **Note:** Multiple client applications can log in using the same `username` argument. However, this approach increases your risk of getting errors due to query limits. A user can have up to 10 query cursors open at a time. If 10 `QueryLocator` cursors are open when a client application, logged in as the same user, attempts to open a new one, then the oldest of the 10 cursors is released. If the client application attempts to open the released query cursor, an error results.

The limit is 3600 calls to `login()` per user per hour. Exceeding this limit results in a "Login Rate Exceeded" error.

Enterprise and Partner Endpoints

In API version 11.1 and earlier, client applications built with the partner WSDL can send requests to the enterprise endpoint and enterprise WSDL applications can send requests to the partner endpoint. Beginning with version 12.0, this functionality is not supported.

Logging In When Using a Proxy

If you log in to Salesforce via a proxy, set the proxy host and port on the instance of the ConnectorConfig class that you use to log in. Optionally, set the username and password if you must authenticate on the proxy.

```
ConnectorConfig config = new ConnectorConfig();
config.setUsername(userId);
config.setPassword(password);
config.setAuthEndpoint(authEndPoint);
config.setProxy(proxyHost, proxyPort);
// Set the username and password if your proxy must be authenticated
config.setProxyUsername(proxyUsername);
config.setProxyPassword(proxyPassword);
try {
    EnterpriseConnection connection = new EnterpriseConnection(config);
    // etc.
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
```

Session Expiration

Client applications do not need to explicitly log out to end a session. Sessions expire automatically after a predetermined length of inactivity, which can be configured in Salesforce from Setup by clicking **Security Controls**. The default is 120 minutes (two hours). If you make an API call, the inactivity timer is reset to zero.

Authenticating Active Self-Service Users

 **Note:** Starting with Spring '12, the Self-Service portal isn't available for new orgs. Existing orgs continue to have access to the Self-Service portal.

To authenticate active Self-Service users, use the [LoginScopeHeader](#) to specify the [Organization](#) ID against which Self-Service users are authenticated. A Self-Service user must exist and be active before being authenticated (see [SelfServiceUser](#)).

Authenticating Customer Community Users in Salesforce Communities

To authenticate an active community user who has the “API Enabled” permission, use the [LoginScopeHeader](#) to specify the [Organization](#) ID of the org with communities. Community users must exist, be active, and belong to communities in the organization before being authenticated.

Client applications can send login requests to one of the following endpoints (using valid values for the authentication endpoint).

Enterprise WSDL:

- String authEndPoint = "https://login.salesforce.com/services/Soap/c/37.0/"
- String authEndPoint = "https://community-domain/path-prefix/Soap/c/37.0/"

Partner WSDL:

- String authEndPoint = "https://login.salesforce.com/services/Soap/u/37.0/"
- String authEndPoint = "https://community-domain/path-prefix/Soap/u/37.0/"

Logging Out

Salesforce recommends that you always call [logout\(\)](#) to end a session when it is no longer needed. This call ends any child sessions in addition to the session being logged out. Logging out instead of waiting for the configured session expiration provides the most protection.

Sample Code—Java

This sample logs a user in with the specified username, password, and authentication endpoint URL. The sample writes user and session information to the console after a successful login. Before running this sample, replace the values for username, password, and authentication endpoint with valid values.

To learn how to generate and import the web service WSDL needed to make API calls, see [Step 2: Generate or Obtain the Web Service WSDL](#) in the Quick Start.

```
public boolean loginSample() {  
    boolean success = false;  
    String username = "username";  
    String password = "password";  
    String authEndPoint = "https://login.salesforce.com/services/Soap/c/24.0/";  
  
    try {  
        ConnectorConfig config = new ConnectorConfig();  
        config.setUsername(username);  
        config.setPassword(password);  
  
        System.out.println("AuthEndPoint: " + authEndPoint);  
        config.setAuthEndpoint(authEndPoint);  
  
        connection = new EnterpriseConnection(config);  
  
        // Print user and session info  
        GetUserInfoResult userInfo = connection.getUserInfo();  
        System.out.println("UserID: " + userInfo.getUserId());  
        System.out.println("User Full Name: " + userInfo.getUserFullName());  
        System.out.println("User Email: " + userInfo.getUserEmail());  
        System.out.println();  
        System.out.println("SessionID: " + config.getSessionId());  
        System.out.println("Auth End Point: " + config.getAuthEndpoint());  
        System.out  
            .println("Service End Point: " + config.getServiceEndpoint());  
        System.out.println();  
  
        success = true;  
    } catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
  
    return success;  
}
```

Sample Code—C#

This sample logs a user in using the specified username and password. The result of the login call contains the service endpoint URL, which is the virtual server instance that's servicing your organization, and a unique session ID. The sample sets these returned values on the binding. It sets the binding URL to the returned service endpoint. It also sets the session ID on the session header that is used on all API calls. Next, the sample writes user and session information to the console after a successful login. Before running this sample, replace the values for user name and password with valid values.

To learn how to generate and import the web service WSDL needed to make API calls, see [Step 2: Generate or Obtain the Web Service WSDL](#) in the Quick Start.

```
public bool loginSample()
{
    Boolean success = false;
    string username = "username";
    string password = "password";

    // Create a service object
    binding = new SforceService();

    LoginResult lr;
    try
    {
        Console.WriteLine("\nLogging in...\n");
        lr = binding.login(username, password);

        /**
         * The login results contain the endpoint of the virtual server instance
         * that is servicing your organization. Set the URL of the binding
         * to this endpoint.
         */
        // Save old authentication end point URL
        String authEndPoint = binding.Url;
        // Set returned service endpoint URL
        binding.Url = lr.serverUrl;

        /** Get the session ID from the login result and set it for the
         * session header that will be used for all subsequent calls.
         */
        binding.SessionHeaderValue = new SessionHeader();
        binding.SessionHeaderValue.sessionId = lr.sessionId;

        // Print user and session info
        GetUserInfoResult userInfo = lr.userInfo;
        Console.WriteLine("UserID: " + userInfo.userId);
        Console.WriteLine("User Full Name: " +
            userInfo.userFullName);
        Console.WriteLine("User Email: " +
            userInfo.userEmail);
        Console.WriteLine();
        Console.WriteLine("SessionID: " +
            lr.sessionId);
        Console.WriteLine("Auth End Point: " +
            authEndPoint);
        Console.WriteLine("Service End Point: " +
            binding.Url);
    }
}
```

```

        lr.serverUrl);
Console.WriteLine();

// Return true to indicate that we are logged in, pointed
// at the right URL and have our security token in place.
success = true;
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
                      e.Message + "\n" + e.StackTrace);
}
return success;
}

```

Arguments

Name	Type	Description
username	string	Login username.
password	string	Login password associated with the specified username.

The login request size is limited to 10 KB.

Response

[LoginResult](#)

Faults

[LoginFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

LoginResult

The [login\(\)](#) call returns a `LoginResult` object, which has the following properties:

Name	Type	Description
metadataServerUrl	string	URL of the endpoint that will process subsequent metadata API calls. Your client application needs to set the endpoint.

Name	Type	Description
passwordExpired	boolean	Indicates whether the password used during the login attempt is expired (<code>true</code>) or not (<code>false</code>). If the password has expired, then the API returns a valid <code>sessionId</code> , but the only allowable operation is the <code>setPassword()</code> call.
serverUrl	string	URL of the endpoint that will process subsequent API calls. Your client application needs to set the endpoint.
sessionId	string	Unique ID associated with this session. Your client application needs to set this value in the session header.
userId	ID	ID of the user associated with the specified username and password.
userInfo	getUserInfoResult	User information fields. For a list of these fields, see getUserInfoResult .

logout()

Ends the session of the logged-in user.

Syntax

```
connection.logout();
```

Usage

This call ends the session for the logged-in user issuing the call. No arguments are needed.

To end one or more sessions started by someone other than the logged-in user, see [invalidateSessions\(\)](#).

Sample Code—Java

This sample calls `logout()` to log the current user out and writes a message to the console.

```
public void logoutSample() {
    try {
        connection.logout();
        System.out.println("Logged out.");
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample calls `logout()` to log the current user out and writes a message to the console.

```
public void logoutSample()
{
```

```
try
{
    binding.logout();
    Console.WriteLine("Logged out.");
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
                      e.Message + "\n" + e.StackTrace);
}
}
```

Arguments

This call uses no arguments. It ends the session for the logged-in user issuing the call, so no arguments are needed. The logged-in user is identified by the [sessionId](#) specified in the [SessionHeader](#) for this call.

Response

Void is returned. Because failure of the call means that the session has already been logged out, no results are needed. Any unexpected error, such as system unavailability, throws an error that should be handled by your client application.

Faults

[UnexpectedErrorFault](#)

merge ()

Merge up to three records into one.

Syntax

```
MergeResult []= connection.merge (MergeRequest [] mergeRequests);
```

Usage

Use this call to merge records of the same object type into one of the records, deleting the others, and re-parenting any related records. Each merge operation is within one transaction. A batch merge has multiple transactions, one for each element in the batch.

The only supported object types are [Lead](#), [Contact](#) and [Account](#).

The [masterRecord](#) field indicates which of the records is the master record that the others are merged into. You can use [queryAll \(\)](#) to view records that have been deleted during a merge.

This call requires that you decide prior to the merge call if there are any field values from the non-master record(s) that should supersede the values in the master record. If so, the field names and their new values should be set in the [masterRecord](#) of the MergeRequest, similar to a call to update.

The following limits apply to any merge request:

- Up to 200 merge requests can be made in a single SOAP call.
- Up to three records can be merged in a single request, including the master record. This is the same limit as the Salesforce user interface. If you have more than three records to merge, use the same master record in each request to avoid errors.
- External ID fields cannot be used with `merge()`.

To find all records that have been merged since a given point in time, you can use `queryAll()` with a SELECT statement similar to the following:

```
SELECT Id FROM Contact WHERE isDeleted=true and masterRecordId != null  
AND SystemModstamp > 2006-01-01T23:01:01+01:00
```

It is a recommended best practice to narrow your result set to the most relevant records by filtering on `SystemModstamp`.

 **Note:** Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the `AllowFieldTruncationHeader` SOAP header.

Sample Code—Java

This sample merges one account with a master account. It creates two accounts and attaches a note to the account to merge with the master. After the merge, it writes the ID of the account that was merged and the number of child records updated, which in this case is one because the note of the merged account is moved to the master account.

```
public Boolean mergeRecords() {  
    Boolean success = false;  
    // Array to hold the results  
    String[] accountIds = new String[2];  
    try {  
        // Create two accounts to merge  
        Account[] accounts = new Account[2];  
        Account masterAccount = new Account();  
        masterAccount.setName("MasterAccount");  
        masterAccount.setDescription("The Account record to merge with.");  
        accounts[0] = masterAccount;  
        Account accountToMerge = new Account();  
        accountToMerge.setName("AccountToMerge");  
        accountToMerge  
            .setDescription("The Account record that will be merged.");  
        accounts[1] = accountToMerge;  
        SaveResult[] saveResults = connection.create(accounts);  
  
        if (saveResults.length > 0) {  
            for (int i = 0; i < saveResults.length; i++) {  
                if (saveResults[i].isSuccess()) {  
                    accountIds[i] = saveResults[i].getId();  
                    System.out.println("Created Account ID: " +  
                        accountIds[i]);  
                } else {  
                    // If any account is not created,  
                    // print the error returned and exit  
                    System.out  
                        .println("An error occurred while creating account."  
                            + " Error message: "  
                            + saveResults[i].getErrors()[0].getMessage());  
                }  
            }  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return success;  
}
```

```
        return success;
    }
}

// Set the IDs of the accounts
masterAccount.setId(accountIds[0]);
accountToMerge.setId(accountIds[1]);

// Attach a note to the account to be merged with the master,
// which will get re-parented after the merge
Note note = new Note();
System.out.println("Attaching note to record " +
    accountIds[1]);
note.setParentId(accountIds[1]);
note.setTitle("Merged Notes");
note.setBody("This note will be moved to the "
    + "MasterAccount during merge");
SaveResult[] sRes = connection.create(new SObject[] { note });
if (sRes[0].isSuccess()) {
    System.out.println("Created Note record.");
} else {
    Error[] errors = sRes[0].getErrors();
    System.out.println("Could not create Note record: "
        + errors[0].getMessage());
}

// Perform the merge
MergeRequest mReq = new MergeRequest();
masterAccount.setDescription("Was merged");
mReq.setMasterRecord(masterAccount);
mReq.setRecordToMergeIds(new String[] { saveResults[1].getId() });
MergeResult mRes = connection.merge(new MergeRequest[] { mReq })[0];

if (mRes.isSuccess())
{
    System.out.println("Merge successful.");
    // Write the IDs of merged records
    for(String mergedId : mRes.getMergedRecordIds()) {
        System.out.println("Merged Record ID: " + mergedId);

    }
    // Write the updated child records. (In this case the note.)
    System.out.println(
        "Child records updated: " + mRes.getUpdatedRelatedIds().length);
    success = true;
} else {
    System.out.println("Failed to merge records. Error message: " +
        mRes.getErrors()[0].getMessage());
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
```

```
    return success;
}
```

Sample Code—C#

This sample merges one account with a master account. It creates two accounts and attaches a note to the account to merge with the master. After the merge, it writes the ID of the account that was merged and the number of child records updated, which in this case is one because the note of the merged account is moved to the master account.

```
public Boolean mergeRecords()
{
    Boolean success = false;
    // Array to hold the results
    String[] accountIds = new String[2];
    try
    {
        // Create two accounts to merge
        Account[] accounts = new Account[2];
        Account masterAccount = new Account();
        masterAccount.Name = "MasterAccount";
        masterAccount.Description = "The Account record to merge with.";
        accounts[0] = masterAccount;
        Account accountToMerge = new Account();
        accountToMerge.Name = "AccountToMerge";
        accountToMerge
            .Description = "The Account record that will be merged.";
        accounts[1] = accountToMerge;
        SaveResult[] saveResults = binding.create(accounts);

        if (saveResults.Length > 0)
        {
            for (int i = 0; i < saveResults.Length; i++)
            {
                if (saveResults[i].success)
                {
                    accountIds[i] = saveResults[i].id;
                    Console.WriteLine("Created Account ID: "
                        + accountIds[i]);
                }
                else
                {
                    // If any account is not created,
                    // print the error returned and exit
                    Console.WriteLine("An error occurred while creating account."
                        + " Error message: "
                        + saveResults[i].errors[0].message);
                }
            }
        }
    }

    // Set the Ids of the accounts
    masterAccount.Id = accountIds[0];
}
```

```
accountToMerge.Id = accountIds[1];

// Attach a note to the account to be merged with the master,
// which will get re-parented after the merge
Note note = new Note();
Console.WriteLine("Attaching note to record " +
    accountIds[1]);
note.ParentId = accountIds[1];
note.Title = "Merged Notes";
note.Body = "This note will be moved to the "
    + "MasterAccount during merge";
SaveResult[] sRes = binding.create(new sObject[] { note });
if (sRes[0].success)
{
    Console.WriteLine("Created Note record.");
}
else
{
    Error[] errors = sRes[0].errors;
    Console.WriteLine("Could not create Note record: "
        + errors[0].message);
}

// Perform the merge
MergeRequest mReq = new MergeRequest();
masterAccount.Description = "Was merged";
mReq.masterRecord = masterAccount;
mReq.recordToMergeIds = new String[] { saveResults[1].id };
MergeResult mRes = binding.merge(new MergeRequest[] { mReq })[0];

if (mRes.success)
{
    Console.WriteLine("Merge successful.");
    // Write the IDs of merged records
    foreach (String mergedId in mRes.mergedRecordIds)
    {
        Console.WriteLine("Merged Record ID: " + mergedId);
    }
    // Write the updated child records. (In this case the note.)
    Console.WriteLine(
        "Child records updated: " + mRes.updatedRelatedIds.Length);
    success = true;
}
else
{
    Console.WriteLine("Failed to merge records. Error message: " +
        mRes.errors[0].message);
}
}

catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
```

```

    return success;
}

```

Arguments

This call accepts an array of MergeRequest objects. A MergeRequest object contains the following properties.

Name	Type	Description
masterRecord	sObject	Required. Must provide the ID of the object that other records will be merged into. Optionally, provide the fields to be updated and their values.
recordToMergeIds	ID[]	Required. Minimum of one, maximum of two. The other record or records to be merged into the master record.

Response

MergeResult []

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

[InvalidIdFault](#)

SEE ALSO:

[API Call Basics](#)

MergeResult

The `merge ()` call returns a MergeResult object, which has the following properties:

Name	Type	Description
errors	Error[]	If an error occurred during the <code>merge ()</code> call, an array of one or more Error objects providing the error code and description.
id	ID	ID of the master record, the record into which the other records were merged.
mergedRecordIds	ID[]	ID of the records that were merged into the master record. If successful, the values will match mergeRequest.recordToMergeIds.
success	boolean	Indicates whether the merge was successful (<code>true</code>) or not (<code>false</code>).
updatedRelatedIds	ID[]	ID of all related records that were moved (re-parented) as a result of the merge, and that are viewable by the user sending the merge call.

performQuickActions ()

Executes quick actions of type create or update.

Syntax

```
PerformQuickActionResult[] = connection.performQuickActions(PerformQuickActionRequest
PerformQuickActionRequest[]);
```

Usage

Use the `performQuickActions()` call to perform a specific quick action. Returns an array of `PerformQuickActionResult` objects.

 **Note:** If you're accessing the API using a custom community URL and you use the `performQuickActions()` call to create a group, the group will only be available within that community.

Sample—Java

This sample uses a quick action to create a new contact.

```
public void example() throws Exception {

    PerformQuickActionRequest req = new PerformQuickActionRequest();

    Contact con = new Contact();
    con.setLastName("Smith");

    req.setQuickActionName("Account.QuickCreateContact");
    req.setParentId("001D00000JSaHa");
/* For version 29.0 and greater, use setContextId */
    req.setRecords(new SObject[] { con }); //you can only save one record here
    PerformQuickActionResult[] pResult =
        conn.performQuickActions(new PerformQuickActionRequest[] { req } );
    for(PerformQuickActionResult pr : pResult) {
        assert pr.getSuccess();
        assert pr.getCreated();
        assert pr.getErrors().length == 0;
        System.out.println("Id of the record created: " + pr.getIds()[0]);
        System.out.println("Id of the feeditem for action performed: " +
            pr.getFeedItemIds()[0]);
    }
}
```

Arguments

Name	Type	Description
quickActions	PerformQuickActionRequest	The action request to perform.

PerformQuickActionRequest

Name	Type	Description
parentOrContextId	ID	<ul style="list-style-type: none"> In API version 28.0 <code>parentID</code> is the ID of the sObject on which to create a record for the request. In API version 29.0 and greater, <code>contextID</code> is the ID of the context on which to create a record for the request.
quickActionName	string	The parent or context sObject and action name—for example, Opportunity.QuickCreateOpp.
records	SObject[]	The record to be created. Only one record can be saved at a time.

Response

[PerformQuickActionResult](#)

PerformQuickActionResult

The `performQuickActions()` call returns an array of PerformQuickActionResult objects.

Name	Type	Description
created	boolean	If <code>true</code> , the record was created successfully and if <code>false</code> , no record was created.
errors	Error[]	If an error occurred during the call, an array of one or more <code>Error</code> objects providing the error information.
feedItemIds	ID[]	Returns an array of unique identifiers of a feed item in the form of a string with IDs; in partner portals, a type with an ID is returned.
ids	ID[]	An array of IDs.
success	boolean	If <code>true</code> , the action executed successfully and if <code>false</code> , the action failed.
successMessage	string	Returns the message that displays to the user upon successful completion of the action.

process()

Submits an array of approval process instances for approval, or processes an array of approval process instances to be approved, rejected, or removed. For more information, see “Set Up an Approval Process” in the Salesforce online help.

Syntax

```
ProcessResult = connection.process( processType processRequest[])
```

processType can be either [ProcessSubmitRequest](#) or [ProcessWorkitemRequest](#)

Usage

Use the `process()` call to perform either of the following two tasks:

- Submit an array of objects to the approval process. Objects cannot already be in an approval process when submitted. Use the [ProcessSubmitRequest](#) signature.
- Process an object that has been submitted to the approval process by performing an approval action (Approve or Reject). Use the [ProcessWorkitemRequest](#) signature.

Requests are processed and a [ProcessResult](#) is returned with the same process instances as sent in the request.

The failure of a particular record will not cause failure of the entire request.

 **Note:** Because you can fire Apex triggers with this call, you may be updating fields that contain strings.

Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the [AllowFieldTruncationHeader](#) SOAP header.

Sample Code—Java

This sample accepts the ID of the sObject to process the approval for and an array containing the IDs of the next approvers. It creates a process approval request and submits it for approval. Finally, it parses the results of the `process()` call.

```
public void processRecords(String id, String[] approverIds) {  
    ProcessSubmitRequest request = new ProcessSubmitRequest();  
    request.setComments("A comment about this approval.");  
    request.setObjectId(id);  
    request.setNextApproverIds(approverIds);  
    try {  
        ProcessResult[] processResults = connection  
            .process(new ProcessSubmitRequest[] { request });  
        for (ProcessResult processResult : processResults) {  
            if (processResult.isSuccess()) {  
                System.out.println("Approval submitted for: " + id + ":");  
                for (int i = 0; i < approverIds.length; i++) {  
                    System.out  
                        .println("\tBy: " + approverIds[i] + " successful.");  
                }  
                System.out.println("Process Instance Status: "  
                    + processResult.getInstanceStatus());  
            } else {  
                System.out.println("Approval submitted for: " + id  
                    + ", approverIds: " + approverIds.toString() + " FAILED.");  
                System.out.println("Error: "  
                    + processResult.getErrors().toString());  
            }  
        }  
    } catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
}
```

Sample Code—C#

This sample accepts the ID of the sObject to process the approval for and an array containing the IDs of the next approvers. It creates a process approval request and submits it for approval. Finally, it parses the results of the `process()` call.

```
public void processRecords(String id, String[] approverIds)
{
    ProcessSubmitRequest request = new ProcessSubmitRequest();
    request.comments = "A comment about this approval.";
    request.objectId = id;
    request.nextApproverIds = approverIds;
    try
    {
        ProcessResult[] processResults = binding.process(
            new ProcessSubmitRequest[] { request });
        foreach (ProcessResult processResult in processResults)
        {
            if (processResult.success)
            {
                Console.WriteLine("Approval submitted for: " + id + ":");
                for (int i = 0; i < approverIds.Length; i++)
                {
                    Console.WriteLine("\tBy: " + approverIds[i] + " successful.");
                }
                Console.WriteLine("Process Instance Status: "
                    + processResult.instanceStatus);
            }
            else
            {
                Console.WriteLine("Approval submitted for: " + id
                    + ", approverIds: " + approverIds.ToString() + " FAILED.");
                Console.WriteLine("Error: "
                    + processResult.errors.ToString());
            }
        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

ProcessSubmitRequest Arguments

Name	Type	Description
objectId	ID	The object to submit for approval, for example, an Account , Contact , or custom object.
nextApproverIds	ID[]	If the process requires specification of the next approval, the ID of the user to be assigned the next request.
comments	string	The comment to add to the history step associated with this request.

ProcessWorkitemRequest Arguments

Name	Type	Description
action	string	For processing an item after being submitted for approval, a string representing the kind of action to take: Approve, Reject, or Remove. Only system administrators can specify Remove. If the Allow submitters to recall approval requests option is selected for the approval process, the submitter can also specify Remove.
nextApproverIds	ID[]	If the process requires specification of the next approval, the ID of the user to be assigned the next request.
comments	string	The comment to add to the history step associated with this request.
workitemId	ID	The ID of the ProcessInstanceWorkitem that is being approved, rejected, or removed.

Response

[ProcessResult\[\]](#)

Faults

[ALREADY_IN_PROCESS](#)

[NO_APPLICABLE_PROCESS](#)

SEE ALSO:

[API Call Basics](#)

ProcessResult

The [process \(\)](#) call returns a ProcessResult object, which has the following properties, depending on the type of call (submit for approval or process object already submitted to for approval):

Name	Type	Description
actorIds	ID[]	IDs of the users who are currently assigned to this approval step.
entityId	ID	The object being processed.
errors	Error[]	The set of errors returned if the request failed.
instanceId	ID	The ID of the ProcessInstance associated with the object submitted for processing.
instanceStatus	string	The status of the current process instance (not an individual object but the entire process instance). The valid values are "Approved," "Rejected," "Removed," or "Pending."
newWorkItemIds	ID[]	Case-insensitive IDs that point to ProcessInstanceWorkitem items (the set of pending approval requests).
success	boolean	true if processing or approval completed successfully.

query()

Executes a query against the specified object and returns data that matches the specified criteria.

Syntax

```
QueryResult = connection.query(string queryString);
```

Usage

Use the `query()` call to retrieve data from an object. When a client application invokes the `query()` call, it passes in a query expression that specifies the object to query, the fields to retrieve, and any conditions that determine whether a given object qualifies. For an extensive discussion about the syntax and rules used for queries, see the [Salesforce SOQL and SOSL Reference Guide](#).

Upon invocation, the API executes the query against the specified object, caches the results of the query on the API, and returns a query response object to the client application. The client application can then use methods on the query response object to iterate through rows in the query response and retrieve information.

Your client application must be logged in with sufficient access rights to query individual objects within the specified object and to query the fields in the specified field list. For more information, see [Factors that Affect Data Access](#).

Certain objects cannot be queried via the API. To query an object via the `query()` call, its object must be configured as queryable. To determine whether an object can be queried, your client application can invoke the `describeSObjects()` call on the object and inspect its `queryable` property.

 **Tip:** If you use the enterprise WSDL, you should not use `describe` to populate a select list. For example, if a system administrator adds a field to the SObject after you consume it, the `describe` call will pull down the field but your toolkit won't know how to serialize it, and your integration may fail.

You can use `queryAll()` to query on all `Task` and `Event` records, archived or not. You can also filter on the `isArchived` field to find only the archived objects. You cannot use `query()`, it automatically filters out all records where `isArchived` is set to `true`. You can insert, update, or delete archived records.

The query result object contains up to 500 rows of data by default. If the query results exceed 500 rows, then the client application uses the `queryMore()` call and a server-side cursor to retrieve additional rows in 500-row chunks. You can increase the default size up to 2,000 in the `QueryOptions` header. For more details see [Change the Batch Size in Queries](#) in the *Force.com SOQL and SOSL Reference*.

Queries that take longer than two minutes to process will be timed out. For timed out queries, the API returns an API fault element of `InvalidQueryLocatorFault`. If a timeout occurs, refactor your query to return or scan a smaller amount of data.

When querying for fields of type Base64 (see [base64](#)), the query response object returns only one record at a time. You cannot alter this by changing the batch size of the `query()` call.

 **Note:** For multicurrency organizations, special handling is required when querying currency fields containing values in different currencies. For example, if a client application is querying `PricebookEntry` objects based on values in the `UnitPrice` field, and if the `UnitPrice` amounts are expressed in different currencies, then the query logic must handle this case correctly. For example, if the query is trying to retrieve the product codes of all products with a unit price greater than or equal to \$10 USD, the query expression might look something like this:

```
SELECT Product2Id, ProductCode, UnitPrice FROM PricebookEntry
WHERE (UnitPrice >= 10 and CurrencyIsoCode='USD')
OR (UnitPrice >= 5.47 and CurrencyIsoCode='GBP')
OR (UnitPrice >= 8.19 and CurrencyIsoCode='EUR')
```

Sample Code—Java

This sample executes a query that fetches the first names and last names of all contacts. It calls `query()` with the query string to get the first batch of records. It then calls `queryMore()` in a loop to get subsequent batches of records until no records are returned. It writes the first and last names of the contacts queried to the console.

```
public void queryRecords() {
    QueryResult qResult = null;
    try {
        String soqlQuery = "SELECT FirstName, LastName FROM Contact";
        qResult = connection.query(soqlQuery);
        boolean done = false;
        if (qResult.getSize() > 0) {
            System.out.println("Logged-in user can see a total of "
                + qResult.getSize() + " contact records.");
            while (!done) {
                SObject[] records = qResult.getRecords();
                for (int i = 0; i < records.length; ++i) {
                    Contact con = (Contact) records[i];
                    String fName = con.getFirstName();
                    String lName = con.getLastName();
                    if (fName == null) {
                        System.out.println("Contact " + (i + 1) + ": " + lName);
                    } else {
                        System.out.println("Contact " + (i + 1) + ": " + fName
                            + " " + lName);
                    }
                }
                if (qResult.isDone()) {
                    done = true;
                } else {
                    qResult = connection.queryMore(qResult.getQueryLocator());
                }
            }
        } else {
            System.out.println("No records found.");
        }
        System.out.println("\nQuery successfully executed.");
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample executes a query that fetches the first names and last names of all contacts. It calls `query()` with the query string to get the first batch of records. It then calls `queryMore()` in a loop to get subsequent batches of records until no records are returned. It writes the first and last names of the contacts queried to the console.

```
public void queryRecords()
{
    QueryResult qResult = null;
    try
```

```
{  
    String soqlQuery = "SELECT FirstName, LastName FROM Contact";  
    qResult = binding.query(soqlQuery);  
    Boolean done = false;  
    if (qResult.size > 0)  
    {  
        Console.WriteLine("Logged-in user can see a total of "  
            + qResult.size + " contact records.");  
        while (!done)  
        {  
            sObject[] records = qResult.records;  
            for (int i = 0; i < records.Length; ++i)  
            {  
                Contact con = (Contact)records[i];  
                String fName = con.FirstName;  
                String lName = con.LastName;  
                if (fName == null)  
                {  
                    Console.WriteLine("Contact " + (i + 1) + ": " + lName);  
                }  
                else  
                {  
                    Console.WriteLine("Contact " + (i + 1) + ": " + fName  
                        + " " + lName);  
                }  
            }  
            if (qResult.done)  
            {  
                done = true;  
            }  
            else  
            {  
                qResult = binding.queryMore(qResult.queryLocator);  
            }  
        }  
    }  
    else  
    {  
        Console.WriteLine("No records found.");  
    }  
    Console.WriteLine("\nQuery successfully executed.");  
}  
catch (SoapException e)  
{  
    Console.WriteLine("An unexpected error has occurred: " +  
        e.Message + "\n" + e.StackTrace);  
}  
}
```

Arguments

Name	Type	Description
queryString	string	Query string that specifies the object to query, the fields to return, and any conditions for including a specific object in the query. For more information, see the Salesforce SOQL and SOSL Reference Guide .

Response

QueryResult

Faults

[MalformedQueryFault](#)
[InvalidSObjectFault](#)
[InvalidFieldFault](#)
[UnexpectedErrorFault](#)

SEE ALSO:

[queryAll\(\)](#)
[queryMore\(\)](#)
[API Call Basics](#)
https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

QueryResult

The [query\(\)](#) call returns a QueryResult object, which has the following properties:

Name	Type	Description
queryLocator	QueryLocator	A specialized string, similar to ID. Used in queryMore() for retrieving subsequent sets of objects from the query results, if applicable. Represents a server-side cursor. Each user can have up to ten query cursors open at a time.
done	boolean	Indicates whether additional rows need to be retrieved from the query results (<code>false</code>) using queryMore() , or not (<code>true</code>). Your client application can use this value as a loop condition while iterating through the query results.
records	sObject[]	Array of sObjects representing individual objects of the specified object and containing data defined in the field list specified in the queryString . For information on queries that use a <code>GROUP BY</code> clause, see AggregateResult .
size	int	Your client application can use this value to determine whether the query retrieved any rows (<code>size > 0</code>) or not (<code>size = 0</code>). Total number of rows retrieved in the query.

AggregateResult

This object contains the results returned by a [query\(\)](#) if the query contains an aggregate function, such as `MAX()`. AggregateResult is an [sObject](#), but unlike other sObject objects such as Contact, it is read-only and it is only used for query results.

The [QueryResult](#) object has a `records` field that is an array of sObject records matching your query. For example, the following query returns an array of Contact records in the `records` field.

```
SELECT Id, LastName  
FROM Contact  
WHERE FirstName = 'Bob'
```

When a SOQL query contains an aggregate function, the results are a set of aggregated data instead of an array of records for a standard object, such as Contact. Therefore, the `records` field returns an array of AggregateResult records.

For more information on aggregate functions, see “Aggregate Functions” in the [Salesforce SOQL and SOSL Reference Guide](#).

Fields

Each AggregateResult object contains a separate field for each of the items in the `SELECT` list. For the enterprise WSDL, retrieve the result for each item by calling `getField()` on an AggregateResult object when using WSC client framework. For the partner WSDL, retrieve the result for each item by calling `getField()` on an sObject object.

See [Sample Code—Java](#) and [Sample Code—C#](#) for examples that work with the enterprise WSDL.

Sample Code—Java

```
public void queryAggregateResult() {  
    try {  
        String groupByQuery = "SELECT Account.Name n, " +  
            "MAX(Amount) max, MIN(Amount) min " +  
            "FROM Opportunity GROUP BY Account.Name";  
        QueryResult qr = connection.query(groupByQuery);  
        if (qr.getSize() > 0) {  
            System.out.println("Query returned " +  
                qr.getRecords().length + " results.");  
        }  
        for (SObject sObj : qr.getRecords()) {  
            AggregateResult result = (AggregateResult) sObj;  
            System.out.println("aggResult.Account.Name: " +  
                result.getField("n"));  
        }  
        System.out.println("aggResult.max: " +  
            result.getField("max"));  
        System.out.println("aggResult.min: " +  
            result.getField("min"));  
        System.out.println();  
    }  
    } else {  
        System.out.println("No results found.");  
    }  
    System.out.println("\nQuery successfully executed.");  
} catch (ConnectionException ce) {
```

```
        ce.printStackTrace();
    }
}
```

Sample Code—C#

```
private void testAggregateResult()
{
    try
    {
        QueryResult qr = null;

        binding.QueryOptionsValue = new QueryOptions();

        String soqlStr = "SELECT Name, " +
                         "MAX(Amount), " +
                         "MIN(Amount) " +
                         "FROM Opportunity " +
                         "GROUP BY Name";

        qr = binding.query(soqlStr);

        if (qr.size > 0)
        {

            for (int i = 0; i < qr.records.Length; i++)
            {

                sforce.AggregateResult ar = (AggregateResult)qr.records[i];

                foreach (XmlElement e in ar.Any)
                    Console.WriteLine(
                        "{0} - {1}",
                        e.LocalName,
                        e.InnerText
                    );
            }
        }
        else
        {
            Console.WriteLine("No records found");
        }
        Console.WriteLine("Query successfully executed.");
    }
    catch (Exception ex)
    {
        Console.WriteLine(
            "\nFailed to execute query successfully." +
            "error message was: \n" +
            ex.Message
        );
    }
}
```

```
    }  
}
```

QueryLocator

In the `QueryResult` object returned by the `query()` call, `queryLocator` contains a value that you will use in the subsequent `queryMore()` call. Note the following guidelines:

- Use a given `queryLocator` value only once. When you pass it in a `queryMore()` call, the API returns a new `queryLocator` in the `QueryResult`.
- `QueryLocator` objects expire automatically after 15 minutes of inactivity.
- A user can have up to 10 query cursors open at a time. If 10 `QueryLocator` cursors are open when a client application, logged in as the same user, attempts to open a new one, then the oldest of the 10 cursors is released. If the client application attempts to open the released query cursor, an error results.



Note: Cursor limits for different Force.com features are tracked separately. For example, you can have 10 query cursors open and 10 Metadata API cursors at the same time.

A `QueryLocator` represents a server-side cursor.

queryAll()

Retrieves data from specified objects, whether or not they have been deleted.

Syntax

```
QueryResult = connection.queryAll(string queryString);
```

Usage

Use `queryAll` to identify the records that have been deleted because of a merge or delete. `queryAll` has read-only access to the field `isDeleted`; otherwise it is the same as `query()`.

To find records that have been deleted (in preparation for undeleting them with the `undelete()` call), specify `isDeleted = true` in the query string, and for merged records, request the `masterRecord`. For example:

```
SELECT id, isDeleted, masterRecordId FROM Account WHERE masterRecordId='10000000000Abc'
```

You can use `queryAll()` to query on all `Task` and `Event` records, archived or not. You can also filter on the `isArchived` field to find only the archived objects. You cannot use `query()` as it automatically filters out all records where `isArchived` is set to `true`. You can update or delete archived records, though you cannot update the `isArchived` field. If you use the API to insert activities that meet the criteria listed below, the activities will be archived during the next run of the archival background process.

Because Salesforce doesn't track changes to external data, `queryAll()` behaves the same as `query()` for external objects.

For additional information about using `queryAll`, see `query()`.

Sample Code—Java

This sample performs a query to get all the accounts, whether they're deleted or not. It sets a custom batch size of 250 records. It fetches all batches of records by calling `queryAll()` the first time and then `queryMore()`. The names and the value of the `isDeleted` fields of all returned accounts are written to the console.

```
public void queryAllRecords() {
    // Setting custom batch size
    connection.setQueryOptions(250);

    try {
        String soqlQuery = "SELECT Name, IsDeleted FROM Account";
        QueryResult qr = connection.queryAll(soqlQuery);
        boolean done = false;
        if (qr.getSize() > 0) {
            System.out.println("Logged-in user can see a total of "
                + qr.getSize()
                + " contact records (including deleted records).");
            while (!done) {
                SObject[] records = qr.getRecords();
                for (int i = 0; i < records.length; i++) {
                    Account account = (Account) records[i];
                    boolean isDel = account.getIsDeleted();
                    System.out.println("Account " + (i + 1) + ": "
                        + account.getName() + " isDeleted = "
                        + account.getIsDeleted());
                }
                if (qr.isDone()) {
                    done = true;
                } else {
                    qr = connection.queryMore(qr.getQueryLocator());
                }
            }
        } else {
            System.out.println("No records found.");
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample performs a query to get all the accounts, whether they're deleted or not. It sets a custom batch size of 250 records. It fetches all batches of records by calling `queryAll()` the first time and then `queryMore()`. The names and the value of the `isDeleted` fields of all returned accounts are written to the console.

```
public void queryAllRecords()
{
    // Setting custom batch size
    QueryOptions qo = new QueryOptions();
    qo.batchSize = 250;
    qo.batchSizeSpecified = true;
```

```

binding.QueryOptionsValue = qo;

try
{
    String soqlQuery = "SELECT Name, IsDeleted FROM Account";
    QueryResult qr = binding.queryAll(soqlQuery);
    Boolean done = false;
    if (qr.size > 0)
    {
        Console.WriteLine("Logged-in user can see a total of "
            + qr.size
            + " contact records (including deleted records).");
        while (!done)
        {
            sObject[] records = qr.records;
            for (int i = 0; i < records.Length; i++)
            {
                Account account = (Account)records[i];
                Boolean isDel = (Boolean)account.IsDeleted;
                Console.WriteLine("Account " + (i + 1) + ": "
                    + account.Name + " isDeleted = "
                    + account.IsDeleted);
            }
            if (qr.done)
            {
                done = true;
            }
            else
            {
                qr = binding.queryMore(qr.queryLocator);
            }
        }
    }
    else
    {
        Console.WriteLine("No records found.");
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

Arguments

Name	Type	Description
queryString	string	Query string that specifies the object to query, the fields to return, and any conditions for including a specific object in the query. For more information, see the Salesforce SOQL and SOSL Reference Guide .

Response

[QueryResult](#)

Faults

[MalformedQueryFault](#)
[InvalidSObjectFault](#)
[InvalidFieldFault](#)
[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)
[queryMore\(\)](#)

queryMore ()

Retrieves the next batch of objects from a [query \(\)](#).

Syntax

```
QueryResult = connection.queryMore( QueryLocator QueryLocator);
```

Usage

Use this call to process [query \(\)](#) calls that retrieve a large number of records (by default, more than 500) in the result set. The [query \(\)](#) call retrieves the first 500 records and creates a server-side cursor that is represented in the `queryLocator` object. The [queryMore \(\)](#) call processes subsequent records in up to 500-record chunks, resets the server-side cursor, and returns a newly generated `QueryLocator`. To iterate through records in the result set, you generally call [queryMore \(\)](#) repeatedly until all records in the result set have been processed (the `Done` flag is `true`). You can change the maximum number of records returned to up to 2,000. See Change the Batch Size in Queries in the [Salesforce SOQL and SOSL Reference Guide](#) for more information.

You can't use [queryMore \(\)](#) if a query includes a `GROUP BY` clause. See `GROUP BY` and [queryMore \(\)](#) in the [Salesforce SOQL and SOSL Reference Guide](#) for more information.



Note: A [queryMore \(\)](#) call on a parent object invalidates all child cursors in the previous result set. If you need the results from the child, you must use [queryMore \(\)](#) on those results before using [queryMore \(\)](#) on the parent results.

When querying external objects, Salesforce Connect accesses the external data in real time via Web service callouts. Each [queryMore \(\)](#) call results in a Web service callout. The batch boundaries and page sizes depend on your adapter and how you set up the external data source.

We recommend the following:

- When possible, avoid paging by filtering your queries of external objects to return fewer rows than the batch size, which by default is 500 rows. Remember, obtaining each batch requires a [queryMore \(\)](#) call, which results in a Web service callout.
- If the external data frequently changes, avoid using [queryMore \(\)](#) calls. If the external data is modified between [queryMore \(\)](#) calls, you can get an unexpected [QueryResult](#).

If the primary or “driving” object for a `SELECT` statement is an external object, `queryMore()` supports only that primary object and doesn’t support subqueries.

By default, the OData 2.0 and 4.0 adapters for Salesforce Connect use client-driven paging. With client-driven paging, OData adapters convert each `queryMore()` call into an OData query that uses the `$skip` and `$top` system query options to specify the batch boundary and page size. These options are similar to using `LIMIT` and `OFFSET` clauses to page through a result set. If you enable server-driven paging on an external data source, Salesforce ignores the requested page sizes, including the default `queryMore()` batch size of 500 rows. The pages returned by the external system determine the batches.

Sample Code—Java

This sample executes a query that fetches the first names and last names of all contacts. It calls `query()` with the query string to get the first batch of records. It then calls `queryMore()` in a loop to get subsequent batches of records until no records are returned. It writes the first and last names of the contacts queried to the console.

```
public void queryRecords() {
    QueryResult qResult = null;
    try {
        String soqlQuery = "SELECT FirstName, LastName FROM Contact";
        qResult = connection.query(soqlQuery);
        boolean done = false;
        if (qResult.getSize() > 0) {
            System.out.println("Logged-in user can see a total of "
                + qResult.getSize() + " contact records.");
            while (!done) {
                SObject[] records = qResult.getRecords();
                for (int i = 0; i < records.length; ++i) {
                    Contact con = (Contact) records[i];
                    String fName = con.getFirstName();
                    String lName = con.getLastName();
                    if (fName == null) {
                        System.out.println("Contact " + (i + 1) + ": " + lName);
                    } else {
                        System.out.println("Contact " + (i + 1) + ": " + fName
                            + " " + lName);
                    }
                }
                if (qResult.isDone()) {
                    done = true;
                } else {
                    qResult = connection.queryMore(qResult.getQueryLocator());
                }
            }
        } else {
            System.out.println("No records found.");
        }
        System.out.println("\nQuery successfully executed.");
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample executes a query that fetches the first names and last names of all contacts. It calls `query()` with the query string to get the first batch of records. It then calls `queryMore()` in a loop to get subsequent batches of records until no records are returned. It writes the first and last names of the contacts queried to the console.

```
public void queryRecords()
{
    QueryResult qResult = null;
    try
    {
        String soqlQuery = "SELECT FirstName, LastName FROM Contact";
        qResult = binding.query(soqlQuery);
        Boolean done = false;
        if (qResult.size > 0)
        {
            Console.WriteLine("Logged-in user can see a total of "
                + qResult.size + " contact records.");
            while (!done)
            {
                sObject[] records = qResult.records;
                for (int i = 0; i < records.Length; ++i)
                {
                    Contact con = (Contact)records[i];
                    String fName = con.FirstName;
                    String lName = con.LastName;
                    if (fName == null)
                    {
                        Console.WriteLine("Contact " + (i + 1) + ": " + lName);
                    }
                    else
                    {
                        Console.WriteLine("Contact " + (i + 1) + ": " + fName
                            + " " + lName);
                    }
                }
                if (qResult.done)
                {
                    done = true;
                }
                else
                {
                    qResult = binding.queryMore(qResult.queryLocator);
                }
            }
        }
        else
        {
            Console.WriteLine("No records found.");
        }
        Console.WriteLine("\nQuery successfully executed.");
    }
    catch (SoapException e)
    {
```

```

        Console.WriteLine("An unexpected error has occurred: " +
                           e.Message + "\n" + e.StackTrace);
    }
}

```

Arguments

Name	Type	Description
queryLocator	QueryLocator	Represents the server-side cursor that tracks the current processing location in the query result set.

Response

QueryResult

Faults

[InvalidQueryLocatorFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[query\(\)](#)

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

QueryResult

The [queryMore \(\)](#) call returns a `QueryResult` object, which has the following properties:

Name	Type	Description
queryLocator	QueryLocator	A specialized string, similar to ID. Used in the subsequent queryMore () call for retrieving sets of objects from the query results, if applicable.
done	boolean	Indicates whether additional rows need to be retrieved from the query results (<code>false</code>) using another queryMore () call, or not (<code>true</code>). Your client application can use this value as a loop condition while iterating through the query results.
records	sObject[]	Array of sObjects representing individual objects of the specified object and containing data defined in the field list specified in the queryString .
size	int	Total number of rows retrieved in the query. Your client application can use this value to determine whether the query retrieved any rows (<code>size != 0</code>) or not (<code>size = 0</code>). When querying external objects, the system may not know the number of rows that are retrieved from the external data source. If this situation occurs, <code>size = -1</code> .



Note: A `queryMore()` call on a parent object invalidates all child cursors in the previous result set. If you need the results from the child, you must use `queryMore()` on those results before using `queryMore()` on the parent results.

QueryLocator

In the `QueryResult` object returned by the `queryMore()` call, `queryLocator` contains a value that you will use in the subsequent `queryMore()` call. Note the following guidelines for using this value:

- Use a `queryLocator` only once. When you pass it in a `queryMore()` call, the API returns a new `queryLocator` in the `QueryResult`.
- The `queryLocator` value expires automatically after 15 minutes of inactivity.
- A user can have up to ten query cursors open at a time. If ten `QueryLocator` cursors are opened when a client application with the same logged-in user attempts to open a new cursor, then the oldest of the ten cursors is released.
- You can't use a custom metadata query as a `queryLocator`.

A `QueryLocator` represents a server-side cursor.



Note: A `queryMore()` call on a parent object invalidates all child cursors in the previous result set. If you need the results from the child, you must use `queryMore()` on those results before using `queryMore()` on the parent results.

retrieve()

Retrieves one or more records based on the specified IDs.

Syntax

```
sObject[] result = connection.retrieve(string fieldList, string sObjectType, ID ids[]);
```

Usage

Use the `retrieve()` call to retrieve individual records from an object. The client application passes the list of fields to retrieve, the object, and an array of record IDs to retrieve. The `retrieve()` call does not return records that have been deleted.

In general, you use `retrieve()` when you know in advance the IDs of the records to retrieve. Use `query()` instead to obtain records when you do not know the IDs or when you want to specify other selection criteria.

Client applications can use `retrieve()` to perform a client-side join. For example, a client application can run a `query()` to obtain a set of `Opportunity` records, iterate through the returned opportunity records, obtain the `accountId` for each opportunity, and then call `retrieve()` to obtain `Account` information for those `accountIds`.

Records for certain objects cannot be retrieved via the API. To retrieve a record via the `retrieve()` call, its object must be configured as retrievable (`retrieveable` is `true`). To determine whether an object can be retrieved, your client application can invoke the `describeSObjects()` call on the object and inspect its `retrievable` property.

Your client application must be logged in with sufficient access rights to retrieve records within the specified object and to retrieve the fields in the specified field list. For more information, see [Factors that Affect Data Access](#).

Sample Code—Java

This sample retrieves the Id, Name, and Website of the specified Account records. It writes the fields of the retrieved records to the console.

```
public void retrieveRecords(String[] ids) {
    try {
        SObject[] sObjects = connection.retrieve("ID, Name, Website",
                                                "Account", ids);
        // Verify that some objects were returned.
        // Even though we began with valid object IDs,
        // someone else might have deleted them in the meantime.
        if (sObjects != null) {
            for (int i = 0; i < sObjects.length; i++) {
                // Cast the SObject into an Account object
                Account retrievedAccount = (Account) sObjects[i];
                if (retrievedAccount != null) {
                    System.out.println("Account ID: " + retrievedAccount.getId());
                    System.out.println("Account Name: " + retrievedAccount.getName());
                    System.out.println("Account Website: "
                        + retrievedAccount.getWebsite());
                }
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample retrieves the Id, Name, and Website of the specified Account records. It writes the fields of the retrieved records to the console.

```
public void retrieveRecords(String[] ids)
{
    try
    {
        sObject[] sObjects = binding.retrieve("ID, Name, Website",
                                              "Account", ids);
        // Verify that some objects were returned.
        // Even though we began with valid object IDs,
        // someone else might have deleted them in the meantime.
        if (sObjects != null)
        {
            for (int i = 0; i < sObjects.Length; i++)
            {
                // Cast the SObject into an Account object
                Account retrievedAccount = (Account)sObjects[i];
                if (retrievedAccount != null)
                {
                    Console.WriteLine("Account ID: " + retrievedAccount.Id);
                    Console.WriteLine("Account Name: " + retrievedAccount.Name);
                    Console.WriteLine("Account Website: "
```

Arguments

Name	Type	Description
fieldList	string	List of one or more fields in the specified object, separated by commas. You must specify valid field names and must have read-level permissions to each specified field. The fieldList defines the ordering of fields in the result .
sObjectType	string	Object from which to retrieve data. The specified value must be a valid object for your organization. For a complete list of objects, see Standard Objects .
ids	ID[]	Array of one or more IDs of the objects to retrieve. You can pass a maximum of 2000 object IDs to the retrieve() call. For information on IDs, see ID Field Type .

Response

Name	Type	Description
result	sObject[]	Array of one or more sObjects representing individual records of the specified object. The number of sObjects returned in the array matches the number of IDs passed into the <code>retrieve()</code> call. If you do not have access to an object or if a passed ID is invalid, the array returns null for that object. For information on IDs, see ID Field Type .

Faults

InvalidSObjectFault

InvalidFieldFault

UnexpectedErrorFault

SEE ALSO:

API Call Basics

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

search()

Executes a text search in your organization's data.

Syntax

```
SearchResult = connection.search(String searchString);
```

Usage

Use [search\(\)](#) to search for records based on a search string. The [search\(\)](#) call supports searching custom objects. For an extensive discussion about the syntax and rules used for text searches, see the [Salesforce SOQL and SOSL Reference Guide](#).

Certain objects cannot be searched via the API, such as [Attachment](#) objects. To search an object via the [search\(\)](#) call, the object must be configured as searchable (`isSearchable` is `true`). To determine whether an object can be searched, your client application can invoke the [describeSObjects\(\)](#) call on the object and inspect its `searchable` property.

Sample Code—Java

This sample makes the `search()` call by passing it a SOSL query, which returns contacts, leads, and accounts whose phone fields contain a specified value. Next, it gets the `sObject` records from the results and stores the records in arrays depending on the record type. Finally, it writes the fields of the returned contacts, leads, and accounts to the console.

```
public void searchSample() {
    try {
        // Perform the search using the SOSL query.
        SearchResult sr = connection.search(
            "FIND {4159017000} IN Phone FIELDS RETURNING "
            + "Contact(Id, Phone, FirstName, LastName), "
            + "Lead(Id, Phone, FirstName, LastName), "
            + "Account(Id, Phone, Name)");

        // Get the records from the search results.
        SearchRecord[] records = sr.getSearchRecords();

        ArrayList<Contact> contacts = new ArrayList<Contact>();
        ArrayList<Lead> leads = new ArrayList<Lead>();
        ArrayList<Account> accounts = new ArrayList<Account>();

        // For each record returned, find out if it's a
        // contact, lead, or account and add it to the
        // appropriate array, then write the records
        // to the console.
        if (records.length > 0) {
            for (int i = 0; i < records.length; i++) {
                SObject record = records[i].getRecord();
                if (record instanceof Contact) {
                    contacts.add((Contact) record);
                } else if (record instanceof Lead) {
                    leads.add((Lead) record);
                } else if (record instanceof Account) {

```

```

        accounts.add((Account) record);
    }
}

System.out.println("Found " + contacts.size() + " contacts.");
for (Contact c : contacts) {
    System.out.println(c.getId() + ", " + c.getFirstName() + ", "
+ c.getLastName() + ", " + c.getPhone());
}
System.out.println("Found " + leads.size() + " leads.");
for (Lead d : leads) {
    System.out.println(d.getId() + ", " + d.getFirstName() + ", "
+ d.getLastName() + ", " + d.getPhone());
}
System.out.println("Found " + accounts.size() + " accounts.");
for (Account a : accounts) {
    System.out.println(a.getId() + ", " + a.getName() + ", "
+ a.getPhone());
}
} else {
    System.out.println("No records were found for the search.");
}
} catch (Exception ce) {
    ce.printStackTrace();
}
}

```

Sample Code—C#

This sample makes the search() call by passing it a SOSL query, which returns contacts, leads, and accounts whose phone fields contain a specified value. Next, it gets the sObject records from the results and stores the records in arrays depending on the record type. Finally, it writes the fields of the returned contacts, leads, and accounts to the console.

```

public void searchSample()
{
try
{
// Perform the search using the SOSL query.
SearchResult sr = binding.search(
"FIND {4159017000} IN Phone FIELDS RETURNING "
+ "Contact(Id, Phone, FirstName, LastName), "
+ "Lead(Id, Phone, FirstName, LastName), "
+ "Account(Id, Phone, Name)");

// Get the records from the search results.
SearchRecord[] records = sr.searchRecords;

List<Contact> contacts = new List<Contact>();
List<Lead> leads = new List<Lead>();
List<Account> accounts = new List<Account>();

// For each record returned, find out if it's a
// contact, lead, or account and add it to the
}

```

```
// appropriate array, then write the records
// to the console.
if (records.Length > 0)
{
    for (int i = 0; i < records.Length; i++)
    {
        sObject record = records[i].record;
        if (record is Contact)
        {
            contacts.Add((Contact)record);
        }
        else if (record is Lead)
        {
            leads.Add((Lead)record);
        }
        else if (record is Account)
        {
            accounts.Add((Account)record);
        }
    }

    Console.WriteLine("Found " + contacts.Count + " contacts.");
    foreach (Contact c in contacts)
    {
        Console.WriteLine(c.Id + ", " +
            c.FirstName + ", " +
            c.LastName + ", " +
            c.Phone);
    }
    Console.WriteLine("Found " + leads.Count + " leads.");
    foreach (Lead d in leads)
    {
        Console.WriteLine(d.Id + ", " +
            d.FirstName + ", " +
            d.LastName + ", " +
            d.Phone);
    }
    Console.WriteLine("Found " + accounts.Count + " accounts.");
    foreach (Account a in accounts)
    {
        Console.WriteLine(a.Id + ", " +
            a.Name + ", " +
            a.Phone);
    }
}
else
{
    Console.WriteLine("No records were found for the search.");
}
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
```

```
}
```

```
}
```

Arguments

Name	Type	Description
search	string	Search string that specifies the text expression to search for, the scope of fields to search, the list of objects and fields to retrieve, and the maximum number of records to return. For more information, see the Salesforce SOQL and SOSL Reference Guide .

Response

SearchResult

Fault

[InvalidFieldFault](#)

[InvalidSObjectFault](#)

[MalformedSearchFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

SearchResult

The describeDataCategoryGroupStructures() call returns a SearchResult object, which has the following properties.

Name	Type	Description
searchRecords	SearchRecord[]	Array of SearchRecord objects, each of which contains an sObject .
searchResultsMetadata	SearchResultsMetadata	Metadata for <code>searchRecords</code> .
queryId	string	Unique identifier for the <code>SOSL</code> search.

undelete()

Undeletes records from the Recycle Bin.

Syntax

```
UndeleteResult[] = connection.undelete(ID[] ids);
```

Usage

Use this call to restore any deleted record that is undeletable. Undeletable records include those in the Recycle Bin. Records can be put in the Recycle Bin as the result of a [merge \(\)](#) or [delete \(\)](#) call. You can identify deleted records, including records deleted as the result of a merge, using the [queryAll \(\)](#) call.

You should verify that a record can be undeleted before attempting to delete it. Some records cannot be undeleted, for example, [Account](#) records can be undeleted, but not [AccountTeamMember](#) records. To verify that a record can be undeleted, check that the value of the [undeletable](#) flag in the [DescribeSObjectResult](#) for that object is set to `true`.

Since a delete call cascade-deletes child records, an undelete call will undelete the cascade-deleted records. For example, deleting an account will delete all the contacts associated with that account.

You can undelete records that were deleted as the result of a merge, but the child objects will have been re-parented, which cannot be undone.



Note: Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the [AllowFieldTruncationHeader](#) SOAP header.

This call supports the [AllOrNoneHeader](#), [AllowFieldTruncationHeader](#), and [CallOptions](#) headers.

Rollback on Error

The [AllOrNoneHeader](#) header allows you to roll back all changes unless all records are processed successfully. This header is available in API version 20.0 and later. Allows a call to roll back all changes unless all records are processed successfully.

Sample Code—Java

This sample calls [queryAll \(\)](#) to get the last five deleted accounts. It then passes the IDs of these accounts to [undelete \(\)](#), which restores these accounts. Finally, it checks the results of the call and writes the IDs of the restored accounts or any errors to the console.

```
public void undeleteRecords() {
    try {
        // Get the accounts that were last deleted
        // (up to 5 accounts)
        QueryResult qResult = connection
            .queryAll("SELECT Id, SystemModstamp FROM "
                + "Account WHERE IsDeleted=true "
                + "ORDER BY SystemModstamp DESC LIMIT 5");

        String[] Ids = new String[qResult.getSize()];
        // Get the IDs of the deleted records
        for (int i = 0; i < qResult.getSize(); i++) {
            Ids[i] = qResult.getRecords()[i].getId();
        }

        // Restore the records
        UndeleteResult[] undelResults = connection.undelete(Ids);
```

```

// Check the results
for (UndeleteResult result : undelResults) {
    if (result.isSuccess()) {
        System.out.println("Undeleted Account ID: " + result.getId());
    } else {
        if (result.getErrors().length > 0) {
            System.out.println("Error message: "
                + result.getErrors()[0].getMessage());
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

```

Sample Code—C#

This sample calls `queryAll()` to get the last five deleted accounts. It then passes the IDs of these accounts to `undelete()`, which restores these accounts. Finally, it checks the results of the call and writes the IDs of the restored accounts or any errors to the console.

```

public void undeleteRecords()
{
    try
    {
        // Get the accounts that were last deleted
        // (up to 5 accounts)
        QueryResult qResult = binding.queryAll(
            "SELECT Id, SystemModstamp FROM "
            +"Account WHERE IsDeleted=true "
            +"ORDER BY SystemModstamp DESC LIMIT 5");

        String[] Ids = new String[qResult.size];
        // Get the IDs of the deleted records
        for (int i = 0; i < qResult.size; i++)
        {
            Ids[i] = qResult.records[i].Id;
        }

        // Restore the records
        UndeleteResult[] undelResults = binding.undelete(Ids);

        // Check the results
        foreach (UndeleteResult result in undelResults)
        {
            if (result.success)
            {
                Console.WriteLine("Undeleted Account ID: " +
                    result.id);
            }
            else
            {

```

```

        if (result.errors.Length > 0)
        {
            Console.WriteLine("Error message: " +
                result.errors[0].message);
        }
    }
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}

```

Arguments

Name	Type	Description
ids	ID[]	IDs of the records to be restored.

Response

[UndeleteResult](#)

Faults

[UnexpectedErrorFault](#)

SEE ALSO:

[delete\(\)](#)

UndeleteResult

The [undelete \(\)](#) call returns an undeleteResult object with the following properties:

Name	Type	Description
Id	ID	ID of the record being undeleted.
success	boolean	Indicates whether the undelete was successful (<code>true</code>) or not (<code>false</code>).
errors	Error[]	If an error occurred during the undelete () call, an array of one or more Error objects providing the error code and description.

update()

Updates one or more existing records in your organization's data.

Syntax

```
SaveResult[] = connection.update(sObject[] sObjects);
```

Usage

Use this call to update one or more existing records, such as accounts or contacts, in your organization's data. The [update\(\)](#) call is analogous to the UPDATE statement in SQL.

Permissions

Your client application must be logged in with sufficient access rights to [update\(\)](#) records objects for the specified object, as well as individual fields inside that object. For more information, see [Factors that Affect Data Access](#).

Special Handling

Certain objects—and certain fields within those objects—require special handling or permissions. For example, you might also need permissions to access an object's parent object. Before you attempt to update a record for a particular object, be sure to read its description in the [Standard Objects](#) and in the Salesforce online help.

Updateable Objects

Certain records cannot be updated via the API. To update a record via the [update\(\)](#) call, its object must be configured as updateable (`updateable` is `true`). To determine whether an object can be updated, your client application can invoke the [describeSObjects\(\)](#) call on the object and inspect its `updateable` property.

Required Fields

When updating required fields, you must supply a value—you cannot set the value to `null`. For more information, see [Required Fields](#).

ID Fields

Fields whose names contain “`Id`” are either that object's primary key (see [ID Field Type](#)) or a foreign key (see [Reference Field Type](#)). Client applications cannot update primary keys, but they can update foreign keys. For example, a client application can update the `OwnerId` of an [Account](#), because `OwnerId` is a foreign key that refers to the user who owns the account record. Use [describeSObjects\(\)](#) to confirm whether the field can be updated.

This call checks a batch for duplicate `Id` values, and if there are duplicates, the first six are processed. For additional duplicate `Id` values, the [SaveResult](#) for those entries is marked with an error similar to the following:

```
Maximum number of duplicate updates in one batch (6 allowed).
```

Automatically Updated Fields

The API updates certain fields automatically, such as `LastModifiedDate`, `LastModifiedById`, and `SystemModstamp`. You cannot explicitly specify these values in your `update()` call.

Resetting Values to null

To reset a field value to `null`, you add the field name to the `fieldsToNull` array in the `sObject`. You cannot set required fields (`nillable` is `false`) to `null`.

Valid Field Values

You must supply values that are valid for the field's data type, such as integers (not alphabetic characters) for integer fields. In your client application, follow the data formatting rules specified for your programming language and development tool (your development tool will handle the appropriate mapping of data types in SOAP messages).

String Values

When storing values in string fields, the API trims any leading and trailing white space. For example, if the value of a name field is entered as " ABC Company ", then the value is stored in the database as "ABC Company".

Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the `AllowFieldTruncationHeader` SOAP header.

Assignment Rules

When updating `Case` or `Lead` objects, your client application can set `AssignmentRuleHeader` options to have the case or lead automatically assigned to one or more users based on assignment rules configured in the Salesforce user interface. For more information, see [Case](#) or [Lead](#).

Maximum Number of Objects Updated

Your client application can change up to 200 records in a single `update()` call. If an update request exceeds 200 records, the entire operation fails.

Rollback on Error

The `AllOrNoneHeader` header allows you to roll back all changes unless all records are processed successfully. This header is available in API version 20.0 and later. Allows a call to roll back all changes unless all records are processed successfully.

Automatic Subscriptions for Chatter Feeds

To subscribe to records they create, users must enable the `Automatically follow records that I create` option in their personal settings. If users have automatic subscriptions enabled, they automatically follow the records they create and see changes to those records in their Chatter feed on the Home tab.

When you update the owner of a record, the new owner is not automatically subscribed to the record, unless the new owner has automatic subscriptions for records enabled in his or her Chatter feed settings. The previous owner is not automatically unsubscribed. If the new owner has automatic subscriptions for records enabled, the new and previous owners both see any changes to the record in their news feed.

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later.

Updating Records for Different Object Types

You can update records for multiple object types, including custom objects, in one call with API version 20.0 and later. For example, you could update a contact and an account in one call. You can update records for up to 10 objects types in one call.

Records are saved in the same order that they are entered in the `sObjects` input array.

Records for different object types are broken into multiple chunks by Salesforce. A chunk is a subset of the `sObjects` input array and each chunk contains records of one object type. Data is committed on a chunk-by-chunk basis. Any Apex triggers related to the records in a chunk are invoked once per chunk. Consider an `sObjects` input array containing the following set of records:

```
account1, account2, contact1, contact2, contact3, case1, account3, account4, contact4
```

Salesforce splits the records into five chunks:

1. account1, account2
2. contact1, contact2, contact3
3. case1
4. account3, account4
5. contact4

Each call can process up to 10 chunks. If the `sObjects` array contains more than 10 chunks, you must process the records in more than one call.



Warning: You can't update records for multiple object types in one call if one of those types is related to a feature in the Setup area in Salesforce. The only exceptions are the following objects:

- Custom settings objects, which are similar to custom objects. For more information, see "Custom Settings" in the Salesforce online help.
- GroupMember
- Group
- User if the following fields are not being updated:
 - UserId
 - IsActive
 - ForecastEnabled
 - IsPortalEnabled
 - Username
 - ProfileId

update() and Foreign Keys

You can use external ID fields as a foreign key, which allows you to update a record and relate it to another existing record in a single step instead of querying the parent record ID first. To do this, set the foreign key to an instance of the parent sObject that has only the external ID field specified. This external ID should match the external ID value on the parent record.

The following Java and C# examples show you how to update an opportunity and relate it to an existing account using a custom external ID field named `MyExtId__c`. Each example has a method that accepts the ID of the opportunity to update. It creates an opportunity sObject and sets its ID field so that the object points to an existing opportunity to be updated, sets a new value for the stage name field, and then sets the external ID field to the account object. It then updates the opportunity. Once the opportunity is updated, the account becomes its parent and the state name is updated.

Java Example

```
public void updateForeignKeySample(String oppId) {
    try {
        Opportunity updateOpportunity = new Opportunity();
        // Point to an existing opportunity to update
        updateOpportunity.setId(oppId);
        updateOpportunity.setStageName("Qualification");

        Account parentAccountRef = new Account();
        parentAccountRef.setMyExtId__c("SAP1111111");
        updateOpportunity.setAccount(parentAccountRef);

        SaveResult[] results = connection
            .update(new SObject[] { updateOpportunity });
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

C# Example

```
public void updateForeignKeySample(String oppId)
{
    try
    {
        Opportunity updateOpportunity = new Opportunity();
        // Point to an existing opportunity to update
        updateOpportunity.Id = oppId;
        updateOpportunity.StageName = "Prospecting";

        Account parentAccountRef = new Account();
        parentAccountRef.MyExtId__c = "SAP1111111";
        updateOpportunity.Account = parentAccountRef;

        SaveResult[] results = binding.update(
            new sObject[] { updateOpportunity });
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

```
    }  
}
```

Basic Steps for Updating Records

Use this process to update records:

1. Determine the ID of each record that you want to `update()`. For example, you might call `query()` to retrieve a set of records (with their IDs), based on specific criteria, that you would want to update. If you know the ID of the record that you want to update, you can call `retrieve()` instead. For information on IDs, see [ID Field Type](#).
2. Create an `sObject` for each record, and populate its fields with the data that you want to update.
3. Construct an `sObject[]` array and populate that array with the records that you want to update.
4. Call `update()`, passing in the `sObject[]` array.
5. Process the results in the `SaveResult[]` object to verify whether the records have been successfully updated.

Sample Code—Java

This sample accepts the IDs of the accounts to update. It creates two account `sObjects`, sets each with one of the passed IDs so that the `sObject` points to an existing account, and sets other fields. It then makes the `update()` call and verifies the results.

```
public void updateRecords(String[] ids) {  
    Account[] updates = new Account[2];  
  
    Account account1 = new Account();  
    account1.setId(ids[0]);  
    account1.setShippingPostalCode("89044");  
    updates[0] = account1;  
  
    Account account2 = new Account();  
    account2.setId(ids[1]);  
    account2.setNumberOfEmployees(1000);  
    updates[1] = account2;  
  
    // Invoke the update call and save the results  
    try {  
        SaveResult[] saveResults = connection.update(updates);  
        for (SaveResult saveResult : saveResults) {  
            if (saveResult.isSuccess()) {  
                System.out.println("Successfully updated Account ID: "  
                    + saveResult.getId());  
            } else {  
                // Handle the errors.  
                // We just print the first error out for sample purposes.  
                Error[] errors = saveResult.getErrors();  
                if (errors.length > 0) {  
                    System.out.println("Error: could not update " + "Account ID "  
                        + saveResult.getId() + ".");  
                    System.out.println("\tThe error reported was: ("  
                        + errors[0].getStatusCode() + ") "  
                        + errors[0].getMessage() + ".");  
                }  
            }  
        }  
    } catch (ConnectionException e) {  
        System.out.println("An error occurred while connecting to the server: " +  
            e.getMessage());  
    }  
}
```

```
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample accepts the IDs of the accounts to update. It creates two account sObjects, sets each with one of the passed IDs so that the sObject points to an existing account, and sets other fields. It then makes the `update()` call and verifies the results.

```
public void updateRecords(String[] ids)
{
    Account[] updates = new Account[2];

    Account account1 = new Account();
    account1.Id = ids[0];
    account1.ShippingPostalCode = "89044";
    updates[0] = account1;

    Account account2 = new Account();
    account2.Id = ids[1];
    account2.NumberOfEmployees = 1000;
    updates[1] = account2;

    // Invoke the update call and save the results
    try
    {
        SaveResult[] saveResults = binding.update(updates);
        foreach (SaveResult saveResult in saveResults)
        {
            if (saveResult.success)
            {
                Console.WriteLine("Successfully updated Account ID: " +
                    saveResult.id);
            }
            else
            {
                // Handle the errors.
                // We just print the first error out for sample purposes.
                Error[] errors = saveResult.errors;
                if (errors.Length > 0)
                {
                    Console.WriteLine("Error: could not update " +
                        "Account ID " + saveResult.id + ".");
                }
                Console.WriteLine("\tThe error reported was: (" +
                    errors[0].statusCode + ") " +
                    errors[0].message + ".");
            }
        }
    }
}
```

```

        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}

```

Arguments

Name	Type	Description
sObjects	sObject[]	Array of one or more records (maximum of 200) to update.

Response

[SaveResult\[\]](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

SaveResult

The [update \(\)](#) call returns an array of SaveResult objects. Each element in the SaveResult array corresponds to the [sObject\[\]](#) array passed as the [sObjects](#) parameter in the [update \(\)](#) call. For example, the object returned in the first index in the SaveResult array matches the object specified in the first index of the [sObject\[\]](#) array.

A SaveResult object has the following properties:

Name	Type	Description
id	ID	ID of an sObject that you successfully updated. If this field contains a value, then the object was updated successfully. If this field is empty, then the object was not updated and the API returned error information instead.
success	boolean	Indicates whether the update () call succeeded (<code>true</code>) or not (<code>false</code>) for this object.

Name	Type	Description
errors	Error[]	If an error occurred during the update () call, an array of one or more Error objects providing the error code and description. If your organization has active duplicate rules and a duplicate is detected, the SaveResult includes an Error with a data type of DuplicateError .

upsert ()

Creates new records and updates existing records; uses a custom field to determine the presence of existing records. In most cases, we recommend that you use [upsert \(\)](#) instead of [create \(\)](#) to avoid creating unwanted duplicate records (idempotent). Available in the API version 7.0 and later. You can process records for one or more than object type in an [create \(\)](#) or [update \(\)](#) call, but all records must have the same object type in an [upsert \(\)](#) call.

 **Note:** Starting with API version 15.0, if you specify a value for a field that contains a string, and the value is too big for the field, the call fails and an error is returned. In previous versions of the API the value was truncated and the call succeeded. If you wish to keep the old behavior with versions 15.0 and later, use the [AllowFieldTruncationHeader](#) SOAP header.

Syntax

```
UpsertResult[] = connection.upsert(String externalIdFieldName, sObject[] sObjects);
```

Usage

Upsert is a merging of the words insert and update. This call is available for objects if the object has an external ID field or a field with the [idLookup](#) field property.

On custom objects, this call uses an indexed custom field called an external ID to determine whether to create a new record or update an existing record. On standard objects, this call can use the name of any field with the [idLookup](#) field property instead of the external ID.

 **Note:** External ID fields cannot be used with [merge \(\)](#).

For more information about adding custom fields, including external ID fields, to objects, see the “Adding Fields” topic in the Salesforce online help.

Using this call can dramatically reduce how many calls you need to make, particularly when:

- You are integrating your organization’s Salesforce data with ERP (enterprise resource planning) systems such as accounting and manufacturing.
- You are importing data and want to prevent the creation of duplicate objects.

If you are upserting a record for an object that has a custom field with both the `External ID` and `Unique` attributes selected (a unique index), you do not need any special permissions, because the `Unique` attribute prevents the creation of duplicates. If you are upserting a record for an object that has the `External ID` attribute selected but not the `Unique` attribute selected, (a non-unique index) your client application must have the permission “View All Data” to execute this call. Having this permission prevents the client application from using [upsert \(\)](#) to insert an accidental duplicate record because it couldn’t see that the record existed.

 **Note:** Matching by external ID is case-insensitive only if the external ID field has the `Unique` attribute and the `Treat "ABC"` and `"abc"` as duplicate values (case insensitive) option selected. These options are selected in the

Salesforce user interface during field creation. If this is the case, "ABC123" is matched with "abc123." Before performing an operation, if you have external ID fields without the case-insensitive option selected, review your external IDs for any values that would be matched if case was not considered. If such values exist, you may want to modify them to make them unique, or select the case-sensitive option for your external ID fields. For more information about field attributes, see "Custom Field Attributes" in the Salesforce online help.

How Upset Chooses to `update()` or `create()`

Upset uses the external ID to determine whether it should create a new record or update an existing one:

- If the external ID is not matched, then a new record is created.
- If the external ID is matched once, then the existing record is updated.
- If the external ID is matched multiple times, then an error is reported.
- When batch updating multiple records where the external ID is the same for two or more records in your batch call, those records will be marked as errors in the [UpsetResult](#) file. The records will be neither created or updated.

Rollback on Error

The [AllOrNoneHeader](#) header allows you to roll back all changes unless all records are processed successfully. This header is available in API version 20.0 and later. Allows a call to roll back all changes unless all records are processed successfully.

Automatic Subscriptions for Chatter Feeds

To subscribe to records they create, users must enable the `Automatically follow records that I create` option in their personal settings. If users have automatic subscriptions enabled, they automatically follow the records they create and see changes to those records in their Chatter feed on the Home tab.

When you update the owner of a record, the new owner is not automatically subscribed to the record, unless the new owner has automatic subscriptions for records enabled in his or her Chatter feed settings. The previous owner is not automatically unsubscribed. If the new owner has automatic subscriptions for records enabled, the new and previous owners both see any changes to the record in their news feed.

A user can subscribe to a record or to another user. Changes to the record and updates from the users are displayed in the Chatter feed on the user's home page, which is a useful way to stay up-to-date with other users and with changes made to records in Salesforce. Feeds are available in API version 18.0 and later.

`upsert()` and Foreign Keys

You can use external ID fields as a foreign key, which allows you to create or update a record and relate it to another existing record in a single step instead of querying the parent record ID first. To do this, set the foreign key to an instance of the parent sObject that has only the external ID field specified. This external ID should match the external ID value on the parent record. Unlike `create()`, the parent record must already exist when using `upsert()` to create or update a child record related by a foreign key.

The following Java and C# examples upsert an opportunity. In this case, the opportunity doesn't exist in the database, so the `upsert()` call will create it. The opportunity references an existing account. Rather than specify the account ID, which would require a separate query to obtain, we specify an external ID for the account, in this example the `MyExtId__c` custom field.

Java Example

```
public void upsertForeignKeySample() {  
    try {
```

```

Opportunity newOpportunity = new Opportunity();
newOpportunity.setName("UpsertOpportunity");
newOpportunity.setStageName("Prospecting");
Calendar dt = connection.getServerTimestamp().getTimestamp();
dt.add(Calendar.DAY_OF_MONTH, 7);
newOpportunity.setCloseDate(dt);
newOpportunity.setMyExtId__c("UPSSERTID001");

// Parent Account record must already exist
Account parentAccountRef = new Account();
parentAccountRef.setMyExtId__c("SAP111111");
newOpportunity.setAccount(parentAccountRef);

SaveResult[] results = connection
    .upsert("MyExtId__c", new SObject[] { newOpportunity });
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

C# Example

```

public void upsertForeignKeySample()
{
    try
    {
        Opportunity newOpportunity = new Opportunity();
        newOpportunity.Name = "UpsertOpportunity";
        newOpportunity.StageName = "Prospecting";
        DateTime dt = (DateTime)binding.getServerTimestamp().timestamp;
        newOpportunity.CloseDate = dt.AddDays(7);
        newOpportunity.CloseDateSpecified = true;
        newOpportunity.MyExtId__c = "UPSSERTID001";

        // Parent Account record must already exist
        Account parentAccountRef = new Account();
        parentAccountRef.MyExtId__c = "SAP111111";
        newOpportunity.Account = parentAccountRef;

        SaveResult[] results = binding
            .upsert("MyExtId", new sObject[] { newOpportunity });
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
                           e.Message + "\n" + e.StackTrace);
    }
}
}

```

upsert() and Polymorphic Foreign Keys

A polymorphic key is an ID that can refer to more than one type of object as a parent. For example, the `OwnerId` field on the `Case` object can reference either a `User` or a `Group`. Similarly, the `WhoID` field on the `Task` object can reference either a `Contact` or a `Lead`.

With the Enterprise WSDL, such polymorphic foreign key fields are defined as a Name `sObject` type, so it is not possible to pass in the correct `sObject` in an `upsert()` call. However, with the Partner WSDL all foreign key fields—including polymorphic ones—are of type `sObject`, which allows `upsert()` calls on such fields.

- !** **Important:** If your organization has SOQL Polymorphism enabled, polymorphic relationship fields point to `sObjects` and not Names in the Enterprise WSDL; therefore, you can pass the correct `sObject` in a call. SOQL Polymorphism is currently available as a Developer Preview. For more information on enabling SOQL Polymorphism for your organization, contact Salesforce

Sample Code—Java

This sample upserts two accounts using a custom external ID field called `MyExtId__c`. The `upsert()` call matches the accounts based on the `MyExtId__c` field in order to determine whether to create or update the accounts. Before running this sample, change the `MyExtId__c` field name to an existing custom ID field name in your org.

```
public void upsertRecords() {
    SObject[] upserts = new Account[2];

    Account upsertAccount1 = new Account();
    upsertAccount1.setName("Begonia");
    upsertAccount1.setIndustry("Education");
    upsertAccount1.setMyExtId__c("1111111111");
    upserts[0] = upsertAccount1;

    Account upsertAccount2 = new Account();
    upsertAccount2 = new Account();
    upsertAccount2.setName("Bluebell");
    upsertAccount2.setIndustry("Technology");
    upsertAccount2.setMyExtId__c("2222222222");
    upserts[1] = upsertAccount2;

    try {
        // Invoke the upsert call and save the results.
        // Use External_Id custom field for matching records.
        UpsertResult[] upsertResults = connection.upsert(
            "MyExtId__c", upserts);
        for (UpsertResult result : upsertResults) {
            if (result.isSuccess()) {
                System.out.println("\nUpert succeeded.");
                System.out.println((result.isCreated() ? "Insert" : "Update")
                    + " was performed.");
                System.out.println("Account ID: " + result.getId());
            } else {
                System.out.println("The Upert failed because: "
                    + result.getErrors()[0].getMessage());
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample upserts two accounts using a custom external ID field called `MyExtId__c`. The `upsert()` call matches the accounts based on the `MyExtId__c` field in order to determine whether to create or update the accounts. Before running this sample, change the `MyExtId__c` field name to an existing custom ID field name in your org.

```
public void upsertRecords()
{
    sObject[] upserts = new Account[2];

    Account upsertAccount1 = new Account();
    upsertAccount1.Name = "Begonia";
    upsertAccount1.Industry = "Education";
    upsertAccount1.MyExtId__c = "1111111111";
    upserts[0] = upsertAccount1;

    Account upsertAccount2 = new Account();
    upsertAccount2 = new Account();
    upsertAccount2.Name = "Bluebell";
    upsertAccount2.Industry = "Technology";
    upsertAccount2.MyExtId__c = "2222222222";
    upserts[1] = upsertAccount2;

    try
    {
        // Invoke the upsert call and save the results.
        // Use External_Id custom field for matching records.
        UpsertResult[] upsertResults =
            binding.upsert("MyExtId__c", upserts);
        foreach (UpserResult result in upsertResults)
        {
            if (result.success)
            {
                Console.WriteLine("\nUpset succeeded.");
                Console.WriteLine(
                    (result.created ? "Insert" : "Update") +
                    " was performed.");
                Console.WriteLine("Account ID: " + result.id);
            }
            else
            {
                Console.WriteLine("The Upset failed because: " +
                    result.errors[0].message);
            }
        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

Arguments

Name	Type	Description
ExternalIDFieldName	string	Contains the name of the field on this object with the external ID field attribute for custom objects or the <code>idLookup</code> field property for standard objects. The <code>idLookup</code> field property is usually on a field that is the object's ID field or name field, but there are exceptions, so check for the presence of the property in the object you wish to <code>upsert()</code> .
sObjects	<code>sObject[]</code>	Array of one or more records (maximum of 200) to create or update.

Response

[UpsertResult\[\]](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[create\(\)](#)

[update\(\)](#)

[API Call Basics](#)

UpsertResult

The `upsert` call returns an array of UpsertResult objects. Each element in the array corresponds to the `sObject[]` array passed as the `sObjects` parameter in the `upsert()` call. For example, the object returned in the first index in the UpsertResult array matches the object specified in the first index of the `sObject[]` array.

An UpsertResult object has the following properties:

Name	Type	Description
created	boolean	Indicates whether the record was created (<code>true</code>) or updated (<code>false</code>).
errors	Error[]	If errors occurred during the call, an array <code>Error</code> objects, providing the error code and description, is returned.
id	ID	If the call succeeded, the field contains the ID of the record that was either updated or created. If there was an error, the field is null. For more information, see ID Field Type .
success	boolean	Indicates whether the call succeeded (<code>true</code>) or not (<code>false</code>) for this object.

CHAPTER 10 Describe Calls

The following table lists supported describe calls in the API in alphabetical order, and provides a brief description for each. Click a call name to see syntax, usage, and more information for that call.

 **Note:** For a list of API utility calls, see [Utility Calls](#), and for a list of general calls (calls that query, retrieve, or modify data), see [Core Calls](#).

Call	Description
describeAllTabs ()	Returns information about all the tabs—including Lightning Page tabs—available to the logged-in user, regardless of whether the user has chosen to hide tabs in his own user interface via the All Tabs (+) tab customization feature.
describeAppMenu ()	Retrieves metadata about items either in the Salesforce1 navigation menu or the Salesforce drop-down app menu.
describeApprovalLayout ()	Retrieves metadata about approval layouts for the specified object type.
describeAvailableQuickActions ()	In API version 28.0, describes details about actions available for a specified parent. In API version 29.0 and greater, describes details about actions available for a specified context.
describeCompactLayouts ()	Retrieves metadata about compact layouts for the specified object type.
describeDataCategoryGroups ()	Retrieves available category groups for entities specified in the request.
describeDataCategoryGroupStructures ()	Retrieves available category groups along with their data category structure for entities specified in the request.
describeFlexiPages ()	Retrieves metadata details about a set of Lightning Pages.
describeGlobal ()	Retrieves a list of available objects for your organization's data.
describeGlobalTheme ()	Returns information about both objects and themes available to the current logged-in user.
describeKnowledge () Call Name on page 2265	Retrieves the Knowledge language settings in the organization.
describeLayout ()	Retrieves metadata about page layouts for the specified object type.
describePrimaryCompactLayouts ()	Retrieves metadata about the primary compact layout for each of the specified object types.
describeQuickActions ()	Retrieves details about specified actions.
describeSearchScopeOrder ()	Retrieves an ordered list of objects in the logged-in user's default global search scope, including any pinned objects in the user's search results page.
describeSObject ()	Retrieves metadata (field list and object properties) for the specified object type. Superseded by describeSObjects () .

Call	Description
<code>describeSObjects ()</code>	An array-based version of describeSObject.
<code>describeSoftphoneLayout ()</code>	Describes the softPhone layout(s) created for an organization.
<code>describeTabs ()</code>	Returns information about the standard and custom apps available to the logged-in user, as listed in the Force.com app menu at the top of the page.
<code>describeTheme ()</code>	Returns information about themes available to the current logged-in user.

Samples

The samples in this section are based on the enterprise WSDL file. They assume that you have already imported the WSDL file and created a connection. To learn how to do so, see the [Quick Start](#) tutorial.

describeAllTabs ()

Returns information about all the tabs—including Lightning Page tabs—available to the logged-in user, regardless of whether the user has chosen to hide tabs in his own user interface via the All Tabs (+) tab customization feature.

Syntax

```
DescribeTab [] = connection.describeAllTabs();
```

Usage

Use the `describeAllTabs ()` call to obtain information about all the tabs that are available to the logged-in user.

Alternately, use `describeTabs ()` if you want information only about the tabs that display in the Salesforce user interface for the logged-in user.

Sample Code—Java

This sample calls `describeAllTabs ()`, which returns an array of `DescribeTab` results.

```
public void describeAllTabsSample() {
    try {
        // Describe tabs
        DescribeTab[] tabs = connection.describeAllTabs();
        System.out.println("There are " + tabs.length +
            " tabs available to you.");

        // Iterate through the returned tabs
        for (int j = 0; j < tabs.length; j++) {
            DescribeTab tab = tabs[j];
            System.out.println("\tTab " + (j + 1) + ":");

            System.out.println("\t\tName: " + tab.getName());
            System.out.println("\t\tAssociated SObject" + tab.getObjectName());
        }
    }
}
```

```
        System.out.println("\t\tLabel: " + tab.getLabel());
        System.out.println("\t\tURL: " + tab.getUrl());
        DescribeColor[] tabColors = tab.getColors();
        // Iterate through tab colors as needed
        DescribeIcon[] tabIcons = tab.getIcons();
        // Iterate through tab icons as needed
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Arguments

None.

Response

[DescribeTab](#)

describeAppMenu ()

Retrieves metadata about items either in the Salesforce1 navigation menu or the Salesforce drop-down app menu. This call is available in API version 29.0 and later.

If you're accessing the API using a custom community URL, the `describeAppMenu ()` call retrieves the tab set associated with the community ID you specify.

Syntax

```
DescribeAppMenuResult describeResult = connection.describeAppMenu(String appMenuType,
String networkId);
```

Code Sample—Java

This code sample shows how to get the menu items from the Salesforce1 navigation menu.

```
public void describeAppMenu() {
    try {
        //The following two lines are equivalent
        DescribeAppMenuResult describe = connection.describeAppMenu("Salesforce1", "");
        DescribeAppMenuResult appMenu = getClient().describeAppMenu(AppMenuType.Salesforce1);

        for (DescribeAppMenuItem menuItem : appMenu.getAppMenuItems()) {
            if (menuItem.getType() == "Tab.apexPage") {
                String visualforceUrl = menuItem.getContent();
            }
        }
    }
}
```

```
        System.out.println("URL to Visualforce page: " + visualforceUrl);

    }

}

} catch (ConnectionException ce) {
    ce.printStackTrace();
}

}
```

Arguments

Name	Type	Description
appMenuType	string	<p>Restricts the menu data returned to the specified menu type.</p> <p>Valid values are:</p> <ul style="list-style-type: none">• AppSwitcher—to retrieve the data from the Salesforce drop-down app menu• Salesforce1—to retrieve the data from the Salesforce1 navigation menu• NetworkTabs—to retrieve the data from a community tab set
networkId	ID	If the <code>appMenuType</code> is set to <code>NetworkTabs</code> , enter the ID of the community to retrieve the tab set from. If <code>appMenuType</code> is not <code>NetworkTabs</code> , this field must be null or empty.

Response

DescribeAppMenuResult

Faults

`InvalidOrNullForRestrictedPicklist`

DescribeAppMenuResult

The `describeAppMenu()` call returns a list of menu items contained in the specified menu type. The following types are available in API version 29.0 and later.

Name	Type	Description
appMenuItems	DescribeAppMenuItem	Array of one or more menu items in the selected menu type.

DescribeAppMenuItem

Each `DescribeAppMenuItem` object has these fields:

Name	Type	Description
colors	DescribeColor[]	Array of color information used for the tab associated with the menu item.
content	string	Information that helps build the menu item. Each menu item has a different type of content for this field. For example, the Salesforce1 app menu type could contain: <ul style="list-style-type: none"> FlexiPage—the ID of the Lightning Page Visualforce tab—the URL to the page, such as /apex/myApexPage. Menu items of types other than these don't use this field.
icons	Describelcon[]	Array of icon information used for the tab associated with the menu item.
label	string	The display label of the menu item.
name	string	API name of the menu item.
type	string	The type of menu item, and its subtype, if any. Possible values for the Salesforce1 menu type are: <ul style="list-style-type: none"> Standard.Dashboards—Dashboards menu item Standard.Feed—Chatter feed menu item Standard.Today—the Today menu item Standard.Tasks—Tasks menu item Tab.apexPage—a Visualforce tab menu item Tab.flexipage—a Lightning Page tab menu item
url	string	The Salesforce URL the menu item should point to. For the Salesforce1 menu type, this field is <code>null</code> for the Dashboards, Feed, Today, Tasks, and Lightning Page menu items.

describeApprovalLayout()

Retrieves metadata about approval layouts for the specified object type.

Syntax

```
DescribeApprovalLayoutResult approvalLayoutResult = connection.describeApprovalLayout(string sObjectType, string[] approvalProcessNames);
```

Usage

Use this call to retrieve information about the approval layout for a given object type. Each approval process has one approval layout.

If you supply a null value for `approvalProcessNames`, all the approval layouts for the object are returned, instead of the approval layout of each specified approval process.

Sample Code—Java

This sample shows how to get the approval layouts of an Account sObject. It calls `describeApprovalLayout()` with the name of the sObject type to describe. After getting the approval layouts, the sample prints the name and fields found for each approval layout.

```
public void describeApprovalLayoutSample() {
    try {
        String objectToDescribe = "Account";
        DescribeApprovalLayoutResult approvalLayoutResult =
            connection.describeApprovalLayout(objectToDescribe, null);
        System.out.print("There are " + approvalLayoutResult.getApprovalLayouts().length);
        System.out.println(" approval layouts for the " + objectToDescribe + " object.");

        // Get all the approval layouts for the sObject
        for (int i = 0; i < approvalLayoutResult.getApprovalLayouts().length; i++) {
            DescribeApprovalLayout aLayout = approvalLayoutResult.getApprovalLayouts()[i];
            System.out.println(" There is an approval layout with name: " + aLayout.getName());

            DescribeLayoutItem[] layoutItems = aLayout.getLayoutItems();
            System.out.print(" There are " + layoutItems.length);
            System.out.println(" fields in this approval layout.");
            for (int j = 0; j < layoutItems.length; j++) {
                System.out.print("This approval layout has a field with name: ");
                System.out.println(layoutItems[j].getLabel());
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Arguments

Name	Type	Description
<code>sObjectType</code>	<code>string</code>	The specified value must be a valid object for your organization. If the object is a person account, specify Account, or if it is a person contact, specify Contact.
<code>approvalProcessNames</code>	<code>string[]</code>	Optional array of the approval process API names to return approval layout metadata for.

Response

[DescribeApprovalLayoutResult](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

DescribeApprovalLayoutResult

The `describeApprovalLayout()` call returns a `DescribeApprovalLayoutResult` object containing top-level record type information about the passed-in `sObjectType`. Your client application can traverse this object to retrieve detailed metadata about the approval layout.

Name	Type	Description
approvalLayouts	DescribeApprovalLayout[]	List of all the approval layouts in use by the object.

DescribeApprovalLayout

Represents an individual item in the [DescribeApprovalLayout](#) list.

Name	Type	Description
id	ID	Unique ID of this ApprovalLayout. For information on IDs, see ID Field Type .
label	string	Label of the approval layout.
layoutItems	DescribeLayoutItem[]	Array of one or more fields assigned to the approval layout.
name	string	API name of the approval layout.

describeAvailableQuickActions()

In API version 28.0, describes details about actions available for a specified parent. In API version 29.0 and greater, describes details about actions available for a specified context.

Syntax

```
DescribeAvailableQuickActionResult [] = connection.describeAvailableQuickActions(string
parentOrContextType );
```

Usage

Use `describeAvailableQuickActions()` to get the list of actions whose parent (API version 28.0) or context (API version 29.0 and greater) entity name is supplied as well as standard and global actions. The `describeAvailableQuickActions()` call uses the parent entity name, such as "Account", or "null" for global actions, or in API version 29.0 and greater, the context, to return an array of `DescribeAvailableQuickActionResult`.

Sample—Java

This sample retrieves and displays the available action information for the Account object.

```
public void example() throws Exception {
    DescribeAvailableQuickActionResult[] aResult =
        conn.describeAvailableQuickActions("Account");
    for(DescribeAvailableQuickActionResult ar : aResult) {
        System.out.println("Action label: " + ar.getLabel());
        System.out.println("Action name: " + ar.getName());
        System.out.println("Action type: " + ar.getType());
    }
}
```

Arguments

Name	Type	Description
parentOrContextType	string	<p>Either a standard or custom object.</p> <ul style="list-style-type: none"> The <code>parentType</code> applies only to API version 28.0. The <code>contextType</code> applies to API version 29.0 and greater.

Response

An array of [DescribeAvailableQuickActionResult](#) objects.

Faults

`connection.exception` errors

DescribeAvailableQuickActionResult

The `describeAvailableQuickActions()` call returns an array of `DescribeAvailableQuickActionResult` objects. In API version 28.0, each `DescribeAvailableQuickActionResult` object represents details about actions available for a specified parent. In API version 29.0 and greater, each `DescribeAvailableQuickActionResult` object represents details about actions available for a specified context.

Name	Type	Description
actionEnumOrId	string	<p>The unique ID for the action. If the action doesn't have an ID, its API name is used.</p> <p>This field is available in API version 35.0 and later.</p>
label	string	The action label.
name	string	The action name.
type	string	<ul style="list-style-type: none"> LogACall

Name	Type	Description
		<ul style="list-style-type: none"> SocialPost Canvas Create VisualforcePage Update

describeCompactLayouts()

Retrieves metadata about compact layouts for the specified object type.

Syntax

```
DescribeCompactLayoutsResult compactLayoutResult = connection.describeCompactLayouts(string
sObjectType, ID[] recordTypeId);
```

Usage

Use this call to retrieve information about the compact layout for a given object type. This call returns metadata about a given compact layout, including the record type mappings. For more information about compact layouts, see the Salesforce online help.

Sample Code—Java

This sample shows how to get the compact layouts of an Account sObject. It calls `describeCompactLayouts()` with the name of the sObject type to describe. After getting the compact layouts, the sample prints the images, fields, and action buttons found for each compact layout. Next, it prints the system default compact layout for the object, then the mapping information of record types to compact layouts.

```
public void testDescribeCompactLayoutsSample() {
    try {
        String objectToDescribe = "Account";
        DescribeCompactLayoutsResult compactLayoutResult = connection
            .describeCompactLayouts(objectToDescribe, null);
        System.out.println("There are " + compactLayoutResult.getCompactLayouts().length
            + " compact layouts for the " + objectToDescribe + " object.");

        // Get all the compact layouts for the sObject
        for (int i = 0; i < compactLayoutResult.getCompactLayouts().length; i++) {
            DescribeCompactLayout cLayout = compactLayoutResult.getCompactLayouts()[i];
            System.out.println(" There is a compact layout with name: " + cLayout.getName());

            DescribeLayoutItem[] fieldItems = cLayout.getFieldItems();
            System.out.println(" There are " + fieldItems.length + " fields in this compact
layout.");
        }

        // Write field items
```

```

        for (int j = 0; j < fieldItems.length; j++) {
            System.out.println(j + " This compact layout has a field with name: " +
fieldItems[j].getLabel());
        }

        DescribeLayoutItem[] imageItems = cLayout.getImageItems();
        System.out.println(" There are " + imageItems.length + " image fields in this
compact layout.");

        // Write the image items
        for (int j = 0; j < imageItems.length; j++) {
            System.out.println(j + " This compact layout has an image field with name:
" + imageItems[j].getLabel());
        }

        DescribeLayoutButton[] actions = cLayout.getActions();
        System.out.println(" There are " + actions.length + " buttons in this compact
layout.");

        // Write the action buttons
        for (int j = 0; j < actions.length; j++) {
            System.out.println(j + " This compact layout has a button with name: " +
actions[j].getLabel());
        }

        System.out.println("This object's default compact layout is: "
+ compactLayoutResult.getDefaultCompactLayoutId());

        RecordTypeCompactLayoutMapping[] mappings =
compactLayoutResult.getRecordTypeCompactLayoutMappings();
        System.out.println("There are " + mappings.length + " record type to compact
layout mapping for the "
+ objectToDescribe + " object.");
        for (int j = 0; j < mappings.length; j++) {
            System.out.println(j + " Record type " + mappings[j].getRecordTypeId()
+ " is mapped to compact layout " +
mappings[j].getCompactLayoutId());
        }
    }

} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

```

Arguments

Name	Type	Description
sObjectType	string	The specified value must be a valid object for your organization. If the object is a person account, specify Account, or if it is a person contact, specify Contact.

Name	Type	Description
recordTypeId	ID[]	Optional parameter that restricts the compact layout data returned to the specified record types.

Response

[DescribeCompactLayoutsResult](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

DescribeCompactLayoutsResult

The `describeCompactLayouts()` call returns a `DescribeCompactLayoutsResult` object containing top-level record type information about the passed-in `sObjectType`, as well as a mapping of record types to compact layouts. Your client application can traverse this object to retrieve detailed metadata about the compact layout.

Name	Type	Description
compactLayouts	DescribeCompactLayout[]	List of all the compact layouts in use by the object.
defaultCompactLayoutId	ID	ID of the primary compact layout assigned to the object.
recordTypeCompactLayoutMappings	RecordTypeCompactLayoutMapping[]	Record type mapping(s) for the object. The compact layouts associated with the object may be mapped to more than one record type.

DescribeCompactLayout

Represents an individual item in the [DescribeCompactLayout](#) list.

Name	Type	Description
actions	DescribeLayoutButton[]	Array of one or more DescribeLayoutButton items assigned to the compact layout. This list is set by Salesforce and is read-only.
fieldItems	DescribeLayoutItem[]	Array of one or more fields assigned to the compact layout.
id	ID	Unique ID of this CompactLayout. For information on IDs, see ID Field Type .
imageItems	DescribeLayoutItem[]	Array of one or more images assigned to the compact layout. This list is set by Salesforce and is read-only.
label	string	Label of the compact layout.

Name	Type	Description
name	string	API name of the compact layout.
objectType	string	The name of the object to which the compact layout is assigned.

RecordTypeCompactLayoutMapping

Represents a single record type mapping in the `recordTypeCompactLayoutMappings` field in a `DescribeCompactLayoutsResult` object. This object is a map of valid `recordTypeId` to `compactLayoutId`.

Name	Type	Description
available	boolean	Indicates whether this record type is available (<code>true</code>) or not (<code>false</code>). Availability is used to display a list of available record types to the user when they are creating a new record.
compactLayoutId	ID	ID of the compact layout associated with this record type.
compactLayoutName	string	API name of the compact layout.
recordTypeName	string	API name of the record type.
recordTypeId	ID	ID of the record type.

describeDataCategoryGroups ()

Retrieves available category groups for objects specified in the request.

Syntax

```
DescribeDataCategoryGroupResult[] = connection.describeDataCategoryGroups() (string[]
sObjectTypes);
```

Usage

Use this call to describe the available category groups for the objects specified in the request. This call can be used with the `describeDataCategoryGroupStructures()` call to describe all the categories available for a specific object. For additional information about data categories, see “Data Categories in Salesforce.com” in the Salesforce online help.

Sample Code—Java

This sample shows how to retrieve the data category groups associated with:

- Salesforce Knowledge articles
- Questions from the Answers feature

It returns the name, label and description of a category group and the name of the associated `sObject` (article or question). It also returns the number of data categories in the data category group.

```
public void describeDataCategoryGroupsSample() {
    try {
        // Make the describe call for data category groups
        DescribeDataCategoryGroupResult[] results =
            connection.describeDataCategoryGroups(new String[] {
                "KnowledgeArticleVersion", "Question"
            });

        // Get the properties of each data category group
        for (int i = 0; i < results.length; i++) {
            System.out.println("sObject: " +
                results[i].getSobject());
            System.out.println("Group name: " +
                results[i].getName());
            System.out.println("Group label: " +
                results[i].getLabel());
            System.out.println("Group description: " +
                (results[i].getDescription()==null? "" :
                results[i].getDescription()));
            System.out.println("Number of categories: " +
                results[i].getCategoryCount());
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to retrieve the data category groups associated with:

- Salesforce Knowledge articles
- Questions from the Answers feature

It returns the name, label and description of a category group and the name of the associated `sObject` (article or question). It also returns the number of data categories in the data category group.

```
public void describeDataCategoryGroups() {
    try {
        // Make the describe call for data category groups
        DescribeDataCategoryGroupResult[] results =
            binding.describeDataCategoryGroups(new String[] {
                "KnowledgeArticleVersion", "Question"
            });

        // Get the properties of each data category group
        for (int i = 0; i < results.Length; i++) {
            Console.WriteLine("sObject: " +
                results[i].sobject);
            Console.WriteLine("Group name: " +
                results[i].name);
            Console.WriteLine("Group label: " +
                results[i].label);
```

```

Console.WriteLine("Group description: " +
(results[i].description==null? "" :
results[i].description));
Console.WriteLine("Number of categories: " +
results[i].categoryCount);
}
} catch (SoapException e) {
Console.WriteLine("An unexpected error has occurred: " +
e.Message + "\n" + e.StackTrace);
}
}
}

```

Arguments

Name	Type	Description
sObjectTypes	string[]	<p>The specified value can be:</p> <ul style="list-style-type: none"> KnowledgeArticleVersion—to retrieve category groups associated with article types. Question—to retrieve category groups associated with questions. <p>For additional information about articles and questions, see “Work with Articles and Translations” and “Answers Overview” in the Salesforce online help.</p>

Response

DescribeDataCategoryGroupResult

Faults

InvalidSObjectFault

UnexpectedErrorFault

DescribeDataCategoryGroupResult

The describeDataCategoryGroups() call returns a DescribeDataCategoryGroupResult object containing the list of the category groups associated with the specified objects.

Name	Type	Description
categoryCount	int	The number of visible data categories in the data category group.
description	string	The description of the data category group.
label	string	Label for the data category group in the Salesforce user interface.
name	string	The unique name used for API access to the data category group .
sobject	string	The object associated with the data category group.

describeDataCategoryGroupStructures()

Retrieves available category groups along with their data category structure for objects specified in the request.

Syntax

```
describeDataCategoryGroupStructures() [] = connection.  
    describeDataCategoryGroupStructures() (DataCategoryGroupSObjectTypePair []  
pairs, boolean topCategoriesOnly)
```

Usage

Use this call to return the visible data category structure for the given object category group pairs. First use `describeDataCategoryGroups()` to find the available category groups for the objects specified. From the returned list, choose the object category group pairs to pass as the input in `describeDataCategoryGroupStructures()`. This call returns all the visible categories and data category structure as output. For additional information about data categories and data category visibility, see "Data Categories in Salesforce.com" and "Data Category Visibility" in the Salesforce online help.

Sample Code—Java

This sample shows how to use sObject and data category group pairs to retrieve data categories for each pair. It calls `describeDataCategoryGroupStructures()` with two pairs, KnowledgeArticleVersion/Regions and Question/Regions, and iterates through the results of this call. It gets the top categories for each result, which is "All", and then gets the first-level child categories. The sample requires that you set up a data category group called *Regions* with some child categories and associate it with a knowledge article and questions. Alternatively, you can replace the data category group name in the sample if you want to use an existing data category group in your org that has a different name.

```
public void describeDataCateogryGroupStructuresSample() {  
    try {  
        // Create the data category pairs  
        DataCategoryGroupSObjectTypePair pair1 =  
            new DataCategoryGroupSObjectTypePair();  
        DataCategoryGroupSObjectTypePair pair2 =  
            new DataCategoryGroupSObjectTypePair();  
        pair1.setSobject("KnowledgeArticleVersion");  
        pair1.setDataCategoryGroupName("Regions");  
        pair2.setSobject("Question");  
        pair2.setDataCategoryGroupName("Regions");  
  
        DataCategoryGroupSObjectTypePair[] pairs =  
            new DataCategoryGroupSObjectTypePair[] {  
                pair1,  
                pair2  
            };  
  
        // Get the list of top level categories using the describe call  
        DescribeDataCategoryGroupStructureResult[] results =  
            connection.describeDataCategoryGroupStructures(
```

```
        pairs,
        false
    );

    // Iterate through each result and get some properties
    // including top categories and child categories
    for (int i = 0; i < results.length; i++) {
        DescribeDataCategoryGroupStructureResult result =
            results[i];
        String sObject = result.getsObject();
        System.out.println("sObject: " + sObject);
        System.out.println("Group name: " + result.getName());
        System.out.println("Group label: " + result.getLabel());
        System.out.println("Group description: " +
            result.getDescription());

        // Get the top-level categories
        DataCategory[] topCategories = result.getTopCategories();

        // Iterate through the top level categories and retrieve
        // some information
        for (int j = 0; j < topCategories.length; j++) {
            DataCategory topCategory = topCategories[j];
            System.out.println("Category name: " +
                topCategory.getName());
            System.out.println("Category label: " +
                topCategory.getLabel());
            DataCategory [] childCategories =
                topCategory.getChildCategories();
            System.out.println("Child categories: ");
            for (int k = 0; k < childCategories.length; k++) {
                System.out.println("\t" + k + ". Category name: " +
                    childCategories[k].getName());
                System.out.println("\t" + k + ". Category label: " +
                    childCategories[k].getLabel());
            }
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample shows how to use sObject and data category group pairs to retrieve data categories for each pair. It calls `describeDataCategoryGroupStructures()` with two pairs, KnowledgeArticleVersion/Regions and Question/Regions, and iterates through the results of this call. It gets the top categories for each result, which is "All", and then gets the first-level child categories. The sample requires that you set up a data category group called *Regions* with some child categories and associate it

with a knowledge article and questions. Alternatively, you can replace the data category group name in the sample if you want to use an existing data category group in your org that has a different name.

```
public void describeDataCategoryGroupStructuresSample() {
    try {
        // Create the data category pairs
        DataCategoryGroupObjectTypePair pair1 =
            new DataCategoryGroupObjectTypePair();
        DataCategoryGroupObjectTypePair pair2 =
            new DataCategoryGroupObjectTypePair();
        pair1.sobject = "KnowledgeArticleVersion";
        //pair1.setDataCategoryGroupName("Regions");
        pair1.dataCategoryGroupName = "KBArticleCategories";
        pair2.sobject = "Question";
        //pair2.setDataCategoryGroupName("Regions");
        pair2.dataCategoryGroupName = "KBArticleCategories";

        DataCategoryGroupObjectTypePair[] pairs =
            new DataCategoryGroupObjectTypePair[] {
                pair1,
                pair2
            };

        // Get the list of top level categories using the describe call
        DescribeDataCategoryGroupStructureResult[] results =
            binding.describeDataCategoryGroupStructures(
                pairs,
                false
            );

        // Iterate through each result and get some properties
        // including top categories and child categories
        for (int i = 0; i < results.Length; i++) {
            DescribeDataCategoryGroupStructureResult result =
                results[i];
            String sObject = result.sobject;
            Console.WriteLine("sObject: " + sObject);
            Console.WriteLine("Group name: " + result.name);
            Console.WriteLine("Group label: " + result.label);
            Console.WriteLine("Group description: " +
                result.description);

            // Get the top-level categories
            DataCategory[] topCategories = result.topCategories;

            // Iterate through the top level categories and retrieve
            // some information
            for (int j = 0; j < topCategories.Length; j++) {
                DataCategory topCategory = topCategories[j];
                Console.WriteLine("Category name: " +
                    topCategory.name);
                Console.WriteLine("Category label: " +
                    topCategory.label);
                DataCategory [] childCategories =
                    topCategory.childCategories;
            }
        }
    }
}
```

Describe Calls

describeDataCategoryGroupStructures()

```
Console.WriteLine("Child categories: ");
for (int k = 0; k < childCategories.Length; k++) {
    Console.WriteLine("\t" + k + ". Category name: " +
        childCategories[k].name);
    Console.WriteLine("\t" + k + ". Category label: " +
        childCategories[k].label);
}
}
}
}
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
```

Arguments

Name	Type	Description
pairs	DataCategoryGroupSObjectTypePair[]	Specifies a category group and an object to query. Visible data categories are retrieved for that object.
topCategoriesOnly	boolean	Indicates whether the call returns only the top (true) or all the categories (false) visible depending on the user's data category group visibility settings. For more information on data category group visibility, see Data Category Visibility in the Salesforce online help.

DataCategoryGroupSObjectTypePair contains the following fields:

Name	Type	Description
dataCategoryGroupName	string	The unique name used for API access to the data category group.
sobject	string	The object associated with the data category group

Response

describeDataCategoryGroupStructures() []

Faults

InvalidSObjectFault
UnexpectedErrorFault

describeDataCategoryGroupStructures()

The describeDataCategoryGroupStructures() call returns an array of DescribeDataCategoryGroupStructureResult objects containing the category groups and categories associated with the specified objects.

Name	Type	Description
description	string	The description of the data category group.
label	string	The label for the data category group in the Salesforce user interface.
name	string	The unique name used for API access to the data category group.
sobject	string	The object associated with the data category group.
topCategories	DataCategory[]	A list of top level categories visible depending on the user's data category group visibility settings. For more information on data category group visibility, see "Data Category Visibility" in the Salesforce online help.

DataCategory

Name	Type	Description
childDataCategories	DataCategory[]	A recursive list of visible sub categories in the data category.
label	string	The label for the data category in the Salesforce user interface.
name	string	The unique name used for API access to the data category.

describeFlexiPages()

Retrieves metadata details about a set of Lightning Pages.

Lightning Pages are used in several places.

- In Salesforce1, a Lightning Page is the home page for an app that appears in the navigation menu.
- In Lightning Experience, Lightning Pages can be used as the home page for an app, and to customize the layout of record pages and the Home page.

 **Note:** These pages are known as FlexiPages in the API, but are referred to as Lightning Pages in the rest of the Salesforce documentation and UI.

Syntax

```
DescribeFlexiPageResult[] = connection.describeFlexiPages(string flexiPageNames[]);
```

Usage

Use this call to retrieve information about the specified Lightning Pages. This call returns metadata about a set of Lightning Pages, such as each Lightning Page's layout and associated QuickActions.

If you supply an empty list for `flexiPageNames`, no Lightning Pages are returned.



Note: This call is an advanced API call that is typically used by partners who have written custom page rendering code for generating output on a mobile device and need to examine the layout details of a Lightning Page before rendering the output.

Sample Code—Java

This sample shows how to get a Lightning Page. It calls `describeFlexiPage()` with the name of a Lightning Page to describe. After getting the Lightning Page, it prints out the regions, components and properties.

```
public void describeFlexiPageSample() {  
    try {  
        // Retrieve a single FlexiPage  
        String flexiPageName = "MyFlexiPage";  
        DescribeFlexiPageResult[] result = null;  
        result = connection.describeFlexiPages(new String[]{flexiPageName});  
        String msg = String.format("There are %s FlexiPages described in the response",  
        result.length);  
        System.out.println(msg);  
        DescribeFlexiPageResult page = result[0];  
  
        // Iterate over the regions of the FlexiPage  
        for (DescribeFlexiPageRegion region : page.getRegions()) {  
            msg = String.format("Region: %s", region.getName());  
  
            // Iterate over the components in each region  
            for (DescribeComponentInstance cmp : region.getComponents()) {  
                String fullComponentName = cmp.getTypeNamespace() + ":" + cmp.getTypeName();  
  
                System.out.println("Component: " + fullComponentName);  
  
                // Iterate over the properties of each component  
                for (DescribeComponentInstanceProperty prop : cmp.getProperties()) {  
                    msg = String.format("Property [%s] has value [%s]", prop.getName(),  
                    prop.getValue());  
                    System.out.println(msg);  
                }  
            }  
        }  
    } catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
}
```

Faults

[UnexpectedErrorFault](#)

Arguments

Name	Type	Description
flexiPages	string[]	An array of Lightning Pages to be retrieved.
contexts	FlexipageContext[]	An array of contexts for the FlexiPages, such as Account or Contact. This argument isn't necessary for pages of type AppPage or HomePage. Available in API version 35.0 or later.

FlexipageContext

Represents the context of the FlexiPage, such as Account or Contact. Not applicable for pages of type AppPage or HomePage.
Available in API version 35.0 or later.

Name	Type	Description
type	FlexipageContextTypeEnum enumeration of type string	The FlexipageContext type. Valid values are: • ENTITYNAME
value	string	Value of the FlexipageContext.

Response

[DescribeFlexiPageResult](#)

DescribeFlexiPageResult

The `describeFlexiPages()` call returns an array of `DescribeFlexiPageResult` objects containing information about the passed in `flexiPages`, including lists of regions and actions associated with each Lightning Page.

 **Note:** These pages are known as FlexiPages in the API, but are referred to as Lightning Pages in the rest of the Salesforce documentation and UI.

Name	Type	Description
id	ID	The ID of the Lightning Page.
label	string	The label of the Lightning Page.
name	string	The unique name of the Lightning Page.
quickActionList	DescribeQuickActionListResult	List of actions associated with the specified Lightning Page.
regions	DescribeFlexiPageRegion[]	List of regions on the specified Lightning Page.

Name	Type	Description
sobjectType	string	<p>The object the FlexiPage is associated with. For Lightning Pages of type <code>AppPage</code> or <code>HomePage</code>, this field is <code>null</code>. Once the value of this field is set, it can't be changed.</p> <p>Available in API version 33.0 or later.</p>
template	string	<p>Required. The template the FlexiPage is associated with.</p> <p>This field is available in API version 33.0 or later.</p>
type	string	<p>Required. The type of the Lightning Page.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • <code>AppPage</code>—A Lightning Page that is used as the home page for a custom app. • <code>CommAppPage</code>—A Lightning Page that is used to represent a custom page, as created in the Community Builder, in Communities. This value is available in API version 37.0 and later. • <code>CommObjectPage</code>—A Lightning Page used to override an object page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>CommQuickActionCreatePage</code>—A Lightning Page used to override a create record page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>CommRecordPage</code>—A Lightning Page used to override a record page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>CommRelatedListPage</code>—A Lightning Page used to override a related list page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>CommSearchResultPage</code>—A Lightning Page used to override a search result page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>CommThemeLayoutPage</code>—A Lightning Page used to override a theme layout page in Lightning Experience, as created in the Community Builder, in Communities. This value is available in API version 38.0 and later. • <code>HomePage</code>—A Lightning Page that is used to override the Home page in Lightning Experience. This value is available in API version 37.0 and later.

Name	Type	Description
		<ul style="list-style-type: none"> • AvailableforMailAppAppPage—An email application pane used to override the default layout for Lightning for Outlook. This value is available in API version 38.0 and later. • RecordPage—A Lightning Page used to override an object record page in Lightning Experience. This value is available in API version 37.0 and later. • UtilityBar—A Lightning Page used as the utility bar in Lightning Experience apps. This value is available in API version 38.0 and later. <p>Available in API version 32.0 and later.</p>

DescribeFlexiPageRegion

Represents a region of a FlexiPage. There is one region per FlexiPage. A region can contain a record list component or a recent items component that can be scoped to a set of entities.

Name	Type	Description
components	DescribeComponentInstance[]	List of components on the specified Lightning Page.
name	string	Unique name of the FlexiPage region.

DescribeComponentInstance

Represents an instance of a component in a [DescribeFlexiPageRegion](#), such as a filter list.

Name	Type	Description
properties	DescribeComponentInstanceProperty[]	Properties of the component instance.
typeName	string	The first piece of the name of the component. For example, the “flexipage” in <code>flexipage:filterListCard</code> .
typeNamespace	string	The second piece of the name of the component. For example, the “filterListCard” in <code>flexipage:filterListCard</code> .

DescribeComponentInstanceProperty

Represents a single property in a [DescribeComponentInstance](#).

Name	Type	Description
name	string	Name of the component instance property. For example, the <code>filterListCard</code> component has a property whose name is "entityNames".
region	DescribeFlexiPageRegion on page 2258	Reserved for future use. This field is available in API version 34.0 or later.
type	string	If this field value is <code>null</code> , then the <code>ComponentInstanceProperty</code> values apply to the Lightning component. If this field value is <code>decorator</code> , then the <code>ComponentInstanceProperty</code> values apply to the <i>component decorator</i> for the Lightning component. The component decorator is a wrapper around a Lightning component. The decorator can apply additional capabilities to the component when it renders on a specific page in Lightning Experience. For example, you can configure a component decorator around a component on the Lightning Experience utility bar to set the component's height or width when opened. The <code>UtilityBar</code> is the only page type that supports component decorators. Valid values are: <ul style="list-style-type: none"> • <code>decorator</code>
value	string	Value of the component instance property. For example, the <code>filterListCard</code> component has an <code>entityNames</code> property whose value is a comma separated list of entity names like ["Account", "Contact", "Opportunity"]

describeGlobal()

Retrieves a list of available objects for your organization's data.

Syntax

```
DescribeGlobalResult = connection.describeGlobal();
```

Usage

Use `describeGlobal()` to obtain a list of available objects for your organization. You can then iterate through this list and use `describeSObjects()` to obtain metadata about individual objects.

Your client application must be logged in with sufficient access rights to retrieve metadata about your organization's data. For more information, see [Factors that Affect Data Access](#).

Sample Code—Java

This sample shows how to perform a global describe. It then retrieves the sObjects from the global describe result and writes their names to the console.

```
public void describeGlobalSample() {
    try {
        // Make the describeGlobal() call
        DescribeGlobalResult describeGlobalResult =
            connection.describeGlobal();

        // Get the sObjects from the describe global result
        DescribeGlobalSObjectResult[] sobjectResults =
            describeGlobalResult.getsObjects();

        // Write the name of each sObject to the console
        for (int i = 0; i < sobjectResults.length; i++) {
            System.out.println(sobjectResults[i].getName());
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to perform a global describe. It then retrieves the sObjects from the global describe result and writes their names to the console.

```
public void describeGlobalSample()
{
    try
    {
        // Make the describeGlobal() call
        DescribeGlobalResult dgr = binding.describeGlobal();

        // Get the sObjects from the describe global result
        DescribeGlobalSObjectResult[] sObjResults = dgr.sobjects;

        // Write the name of each sObject to the console
        for (int i = 0; i < sObjResults.Length; i++)
        {
            Console.WriteLine(sObjResults[i].name);
        }
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

Arguments

None.

Response

[DescribeGlobalResult](#)

Fault

[UnexpectedErrorFault](#)

SEE ALSO:

[describeSObjects\(\)](#)

[API Call Basics](#)

[Using the Partner WSDL](#)

https://developer.salesforce.com/page/Sample_SOAP_Messages

DescribeGlobalResult

The `describeGlobal()` call returns a `DescribeGlobalResult` object, which has the following properties.

Name	Type	Description
encoding	<code>string</code>	Specifies how an org's data is encoded, such as <code>UTF-8</code> or <code>ISO-8859-1</code> .
maxBatchSize	<code>int</code>	Maximum number of records allowed in a <code>create()</code> , <code>update()</code> , or <code>delete()</code> call.
sobjects	<code>DescribeGlobalSObjectResult[]</code>	List of result objects that returns information about the available objects for your org. Available in API version 17.0 and later. This property enhances the information that was previously available in the <code>types</code> property.
types	<code>string[]</code>	<p>List of available objects for your org. You iterate through this list to retrieve the object string that you pass to <code>describeSObjects()</code>. Beginning with API version 17.0, this property is no longer supported. Use the <code>name</code> property in <code>DescribeGlobalSObjectResult</code> instead.</p>

DescribeGlobalSObjectResult

Represents the properties for one of the objects available for your org. Each object has the following properties:

Name	Type	Description
activateable	<code>boolean</code>	Reserved for future use.

Name	Type	Description
createable	boolean	Indicates whether the object can be created via the <code>create()</code> call (<code>true</code>) or not (<code>false</code>).
custom	boolean	Indicates whether the object is a custom object (<code>true</code>) or not (<code>false</code>).
customSetting	boolean	Indicates whether the object is a custom setting object (<code>true</code>) or not (<code>false</code>).
deletable	boolean	Indicates whether the object can be deleted via the <code>delete()</code> call (<code>true</code>) or not (<code>false</code>).
deprecatedAndHidden	boolean	Reserved for future use.
feedEnabled	boolean	Indicates whether Chatter feeds are enabled for the object (<code>true</code>) or not (<code>false</code>). This property is available in API version 19.0 and later.
keyPrefix	string	<p>Three-character prefix code in the object ID. Object IDs are prefixed with three-character codes that specify the type of the object. For example, <code>Account</code> objects have a prefix of <code>001</code> and <code>Opportunity</code> objects have a prefix of <code>006</code>. Note that a key prefix can sometimes be shared by multiple objects so it does not always uniquely identify an object.</p> <p>Use the value of this field to determine the object type of a parent in those cases where the child may have more than one object type as parent (polymorphic). For example, you may need to obtain the <code>keyPrefix</code> value for the parent of a <code>Task</code> or <code>Event</code>.</p>
label	string	Label text for a tab or field renamed in the user interface, if applicable, or the object name, if not. For example, an organization representing a medical vertical might rename <code>Account</code> to <code>Patient</code> . Tabs and fields can be renamed in the Salesforce user interface. See the Salesforce online help for more information.
labelPlural	string	Label text for an object that represents the plural version of an object name, for example, "Accounts."
layoutable	boolean	Indicates whether the object supports the <code>describeLayout()</code> call (<code>true</code>) or not (<code>false</code>).
mergeable	boolean	Indicates whether the object can be merged with other objects of its type (<code>true</code>) or not (<code>false</code>). <code>true</code> for leads, contacts, and accounts.
mruEnabled	boolean	Indicates whether Most Recently Used (MRU) list functionality is enabled for the object (<code>true</code>) or not (<code>false</code>).
name	string	Name of the object. This name is equivalent to an entry in the <code>types</code> list that is no longer supported, beginning with API version 17.0.
queryable	boolean	Indicates whether the object can be queried via the <code>query()</code> call (<code>true</code>) or not (<code>false</code>).
replicable	boolean	Indicates whether the object can be replicated via the <code>getUpdated()</code> and <code>getDeleted()</code> calls (<code>true</code>) or not (<code>false</code>).
retrieveable	boolean	Indicates whether the object can be retrieved via the <code>retrieve()</code> call (<code>true</code>) or not (<code>false</code>).

Name	Type	Description
searchable	boolean	Indicates whether the object can be searched via the search() call (<code>true</code>) or not (<code>false</code>).
triggerable	boolean	Indicates whether the object supports Apex triggers.
undeletable	boolean	Indicates whether an object can be undeleted using the undelete() call (<code>true</code>) or not (<code>false</code>).
updateable	boolean	Indicates whether the object can be updated via the update() call (<code>true</code>) or not (<code>false</code>).

describeGlobalTheme ()

Returns information about both objects and themes available to the current logged-in user.

Syntax

```
DescribeGlobalTheme = connection.describeGlobalTheme();
```

Usage

Use `describeGlobalTheme()` to get both a list of available objects and theme information about those objects for your organization. `describeGlobalTheme()` is a combination of [describeGlobal\(\)](#) and [describeTheme\(\)](#) combined into a single call.

Your client application must be logged in with sufficient access rights to retrieve theme and object information about your organization's data. For more information, see [Factors that Affect Data Access](#).

`describeGlobalTheme()` is available in API version 29.0 and later.

Sample

This Java sample calls `describeGlobalTheme()` and then iterates over the retrieved object and theme information.

```
public static void describeGlobalThemeExample() {
    try {
        // Get current theme and object information
        DescribeGlobalTheme globalThemeResult = connection.describeGlobalTheme();
        DescribeGlobalResult globalResult = globalThemeResult.getGlobal();
        DescribeThemeResult globalTheme = globalResult.getTheme();
        // For the themes, get the array of theme items, one per object
        DescribeThemeItem[] themeItems = globalTheme.getThemeItems();
        for (int i = 0; i < themeItems.length; i++) {
            DescribeThemeItem themeItem = themeItems[i];
            System.out.println("Theme information for object " + themeItem.getName());
            // Get color and icon info for each themeItem
            DescribeColor colors[] = themeItem.getColors();
            System.out.println("    Number of colors: " + colors.length);
```

```
        int k;
        for (k = 0; k < colors.length; k++) {
            DescribeColor color = colors[k];
            System.out.println("      For Color #" + k + ":");
            System.out.println("          Web RGB Color: " + color.getColor());
            System.out.println("          Context: " + color.getContext());
            System.out.println("          Theme: " + color.getTheme());
        }
        DescribeIcon icons[] = themeItem.getIcons();
        System.out.println("      Number of icons: " + icons.length);
        for (k = 0; k < icons.length; k++) {
            DescribeIcon icon = icons[k];
            System.out.println("      For Icon #" + k + ":");
            System.out.println("          ContentType: " + icon.getContentType());
            System.out.println("          Height: " + icon.getHeight());
            System.out.println("          Theme: " + icon.getTheme());
            System.out.println("          URL: " + icon.getUrl());
            System.out.println("          Width: " + icon.getWidth());
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
```

Response

DescribeGlobalTheme

Faults

UnexpectedErrorFault

SEE ALSO:

DescribeGlobalTheme

DescribeThemeResult

DescribeTheMenuItem

DescribeColor

Describelcon

DescribeGlobalTheme

The `describeGlobalTheme()` call returns `DescribeGlobalTheme`, which contains a `DescribeThemeResult` and a `DescribeGlobalResult`.

Name	Type	Description
global	DescribeGlobalResult	Object information.
theme	DescribeThemeResult	Theme information.

describeKnowledge() Call Name

Retrieves the Knowledge language settings in the organization.

Syntax

```
KnowledgeSettings result = _connection.describeKnowledgeSettings();
```

Usage

Use this call to describe the existing Knowledge language settings, including the default Knowledge language, supported languages, and a list of Knowledge language information. You can also use KnowledgeSettings in the Metadata API to obtain similar information.

Sample Code—Java

This sample shows how to retrieve the Knowledge language settings. It returns the default Knowledge language, a list of Knowledge supported language, including the language code and whether is an active Knowledge language.

```
public void describeKnowledgeSettingsSample() {
    try {

        // Make the describe call for KnowledgeSettings
        KnowledgeSettings result = connection.describeKnowledgeSettings();

        // Get the properties of KnowledgeSettings
        System.out.println("Knowledge default language: " + result.getDefaultLanguage());
        for (KnowledgeLanguageItem lang : result.getLanguages()) {
            System.out.println("Language: " + lang.getName());
            System.out.println("Active: " + lang.isActive());
        }
    } catch (ConnectionException ex) {
        ex.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to retrieve the Knowledge language settings. It returns the default Knowledge language, a list of Knowledge supported language, including the language code and whether is an active Knowledge language.

```
public void describeKnowledgeSettingsSample() {
    try {

        // Make the describe call for KnowledgeSettings
        KnowledgeSettings result = connection.describeKnowledgeSettings();

        // Get the properties of KnowledgeSettings
        Console.WriteLine("Knowledge default language: " + result.getDefaultLanguage());
        for (KnowledgeLanguageItem lang : result.getLanguages()) {
            Console.WriteLine("Language: " + lang.getName());
        }
    }
}
```

```
Console.WriteLine("Active: " + lang.isActive());
}
} catch (SoapException ex) {
ex.printStackTrace();
}
}
```

Response

KnowledgeSettings

describeLayout ()

Retrieves metadata about page layouts for the specified object type.

Syntax

```
DescribeLayoutResult = connection.describeLayout(string sObjectType, string layoutName,
ID recordTypeID[]);
```

Usage

Use this call to retrieve information about the layout (presentation of data to users) for a given object type. This call returns metadata about a given page layout, such as the detail page layout, the edit page layout, and the record type mappings. For additional information, see “Page Layouts” in the Salesforce online help.

Generally, user profiles have one layout associated with each object. In Enterprise, Unlimited, and Performance Editions, user profiles can have multiple layouts per object, where each layout is specific to a given record type. This call returns metadata for multiple layouts, if applicable.

Layouts can be further customized in standard objects that have defined named layouts, which are separate from the primary layout for both the profile and the record type. One example of named layouts is the UserAlt layout defined on the User object, which is consumed in Salesforce1 instead of the primary User layout. New layout names can only be defined by Salesforce, but customization of named layouts is controlled by administrators in the same way as primary layouts.

If you supply a null value for `recordTypeIds`, all the layouts for that user are returned, instead of just the layouts for each specified record type. The same layout may be associated with multiple record types for the user’s profile, in which case there would only be one layout returned.



Note: This call is an advanced API call that is typically used only by partners who have written custom page rendering code for generating output on a specialized device (for example, on PDAs) and need to examine the layout details of an object before rendering the page output.

Use the following procedure to describe layouts:

1. To display a detail page or edit page for a record that already exists, a client application first gets the `recordTypeIds` from the record, then it finds the layoutId associated with that `recordTypeIds` (through `recordTypeMapping`), and finally it uses that layout information to render the page.

2. To display the create version of an edit page, a client application first determines whether more than one record type is available and, if so, presents the user with a choice. Once a record type has been chosen, then the client application uses the layout information to render the page. It uses the picklist values from the [RecordTypeMapping](#) to display valid picklist values for picklist fields.
3. A client application can access the labels for the layout, using the [DescribeLayoutResult](#).

The following restrictions apply to person account record types:

- [describeLayout\(\)](#) for version 7.0 and below will return the default business account record type as the default record type even if the tab default is a person account record type. In version 8.0 and after, it will always be the tab default.
- [describeLayout\(\)](#) for version 7.0 and below will not return any person account record types.

For more information about person account record types, see [Person Account Record Types](#).

Sample Code—Java

This sample shows how to get the layouts of an Account sObject. It calls `describeLayout()` with the name of the sObject type to describe. It doesn't specify record type IDs as a third argument, which means that layouts for all record types will be returned if record types are defined in your org for the specified sObject. After getting the layout, the sample writes the number of detail and edit sections found and their headings. Next, it iterates through each edit layout section and retrieves its components.

```
public void describeLayoutSample() {
    try {
        String objectToDescribe = "Account";
        DescribeLayoutResult dlr =
            connection.describeLayout(objectToDescribe, null, null);
        System.out.println("There are " + dlr.getLayouts().length +
            " layouts for the " + objectToDescribe + " object.");
    }

    // Get all the layouts for the sObject
    for(int i = 0; i < dlr.getLayouts().length; i++) {
        DescribeLayout layout = dlr.getLayouts()[i];
        DescribeLayoutSection[] detailLayoutSectionList =
            layout.getDetailLayoutSections();
        System.out.println(" There are " +
            detailLayoutSectionList.length +
            " detail layout sections");
        DescribeLayoutSection[] editLayoutSectionList =
            layout.getEditLayoutSections();
        System.out.println(" There are " +
            editLayoutSectionList.length +
            " edit layout sections");

        // Write the headings of the detail layout sections
        for(int j = 0; j < detailLayoutSectionList.length; j++) {
            System.out.println(j +
                " This detail layout section has a heading of " +
                detailLayoutSectionList[j].getHeading());
        }

        // Write the headings of the edit layout sections
        for(int x = 0; x < editLayoutSectionList.length; x++) {
            System.out.println(x +
                " This edit layout section has a heading of " +
```

```
        editLayoutSectionList[x].getHeading());
    }

    // For each edit layout section, get its details.
    for(int k = 0; k < editLayoutSectionList.length; k++) {
        DescribeLayoutSection els =
            editLayoutSectionList[k];
        System.out.println("Edit layout section heading: " +
            els.getHeading());
        DescribeLayoutRow[] dlrList = els.getLayoutRows();
        System.out.println("This edit layout section has " +
            dlrList.length + " layout rows.");
        for(int m = 0; m < dlrList.length; m++) {
            DescribeLayoutRow lr = dlrList[m];
            System.out.println(" This row has " +
                lr.getNumItems() + " layout items.");
            DescribeLayoutItem[] dliList = lr.getLayoutItems();
            for(int n = 0; n < dliList.length; n++) {
                DescribeLayoutItem li = dliList[n];
                if ((li.getLayoutComponents() != null) &&
                    (li.getLayoutComponents().length > 0)) {
                    System.out.println("\tLayout item " + n +
                        ", layout component: " +
                        li.getLayoutComponents()[0].getValue());
                }
                else {
                    System.out.println("\tLayout item " + n +
                        ", no layout component");
                }
            }
        }
    }

    // Get record type mappings
    if (dlr.getRecordTypeMappings() != null) {
        System.out.println("There are " +
            dlr.getRecordTypeMappings().length +
            " record type mappings for the " +
            objectToDescribe + " object");
    }
    else {
        System.out.println(
            "There are no record type mappings for the " +
            objectToDescribe + " object.");
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample shows how to get the layouts of an Account sObject. It calls `describeLayout()` with the name of the sObject type to describe. It doesn't specify record type IDs as a third argument, which means that layouts for all record types will be returned if record types are defined in your org for the specified sObject. After getting the layout, the sample writes the number of detail and edit sections found and their headings. Next, it iterates through each edit layout section and retrieves its components.

```
public void describeLayoutSample()
{
    try
    {
        String objectToDescribe = "Account";
        DescribeLayoutResult dlr =
            binding.describeLayout(objectToDescribe, null, null);
        Console.WriteLine("There are " + dlr.layouts.Length +
            " layouts for the " + objectToDescribe + " object.");
    }

    // Get all the layouts for the sObject
    for (int i = 0; i < dlr.layouts.Length; i++)
    {
        DescribeLayout layout = dlr.layouts[i];
        DescribeLayoutSection[] detailLayoutSectionList =
            layout.detailLayoutSections;
        Console.WriteLine(" There are " +
            detailLayoutSectionList.Length +
            " detail layout sections");
        DescribeLayoutSection[] editLayoutSectionList =
            layout.editLayoutSections;
        Console.WriteLine(" There are " +
            editLayoutSectionList.Length +
            " edit layout sections");

        // Write the headings of the detail layout sections
        for (int j = 0; j < detailLayoutSectionList.Length; j++)
        {
            Console.WriteLine(j +
                " This detail layout section has a heading of " +
                detailLayoutSectionList[j].heading);
        }

        // Write the headings of the edit layout sections
        for (int x = 0; x < editLayoutSectionList.Length; x++)
        {
            Console.WriteLine(x +
                " This edit layout section has a heading of " +
                editLayoutSectionList[x].heading);
        }

        // For each edit layout, get its details.
        for (int k = 0; k < editLayoutSectionList.Length; k++)
        {
            DescribeLayoutSection els =
                editLayoutSectionList[k];
        }
    }
}
```

```
Console.WriteLine("Edit layout section heading: " +
    els.heading);
DescribeLayoutRow[] dlrList = els.layoutRows;
Console.WriteLine("This edit layout section has " +
    dlrList.Length + " layout rows.");
for (int m = 0; m < dlrList.Length; m++)
{
    DescribeLayoutRow lr = dlrList[m];
    Console.WriteLine(" This row has " +
        lr.numItems + " layout items.");
    DescribeLayoutItem[] dliList = lr.layoutItems;
    for (int n = 0; n < dliList.Length; n++)
    {
        DescribeLayoutItem li = dliList[n];
        if ((li.layoutComponents != null) &&
            (li.layoutComponents.Length > 0))
        {
            Console.WriteLine("\tLayout item " + n +
                ", layout component: " +
                li.layoutComponents[0].value);
        }
        else
        {
            Console.WriteLine("\tLayout item " + n +
                ", no layout component");
        }
    }
}

// Get record type mappings
if (dlr.recordTypeMappings != null)
{
    Console.WriteLine("There are " +
        dlr.recordTypeMappings.Length +
        " record type mappings for the " +
        objectToDescribe + " object");
}
else
{
    Console.WriteLine(
        "There are no record type mappings for the " +
        objectToDescribe + " object.");
}
}

catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
```

Arguments

Name	Type	Description
sObjectType	string	The specified value must be a valid object for your organization. If the object is a person account, specify Account, or if it is a person contact, specify Contact.
layoutName	string	The specified value must be a valid named layout for this object. Layout names are obtained from <code>namedLayoutInfos</code> in <code>DescribeSObjectResult</code> . The entity name is not valid because the primary layout is not considered "named."
recordTypeIds	ID[]	<p>Optional parameter restricts the layout data returned to the specified record types.</p> <p>To retrieve the layout for the master record type, specify the value <code>012000000000000AAA</code> for the <code>recordTypeIds</code> regardless of the object. This value is returned in the <code>recordTypeInfos</code> for the master record type in the <code>DescribeSObjectResult</code>. Note that a SOQL query returns a null value, not <code>012000000000000AAA</code>.</p> <p>For information on IDs, see ID Field Type.</p>

Response

[DescribeLayoutResult](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

https://developer.salesforce.com/page/Sample_SOAP_Messages

DescribeLayoutResult

The `describeLayout()` call returns a `DescribeLayoutResult` object containing top-level record type information about the passed-in `sObjectType`, as well as a mapping of record types to layouts. Your client application can traverse this object to retrieve detailed metadata about the layout.

 **Tip:** If you have actions in the publisher enabled in your organization, you can retrieve the layout definition for a global publisher layout by using `Global` as the `sObjectType` and `null` as the `recordTypeId`.

Name	Type	Description
feedView	DescribeLayoutFeedView []	Feed view related layout data for a feed-based layout. This field is null for page layouts that are not feed-based.
layouts	DescribeLayout []	Layout(s) associated with the specified <code>sObjectType</code> . In general, there is a one-to-one correspondence between

Name	Type	Description
		layouts and objects. However, in some cases, an object will have multiple layouts in the context of a given user profile.
recordTypeMappings	RecordTypeMapping[]	Record type mapping(s) available for the user. The objects on a user profile can have multiple record types. All record types are returned, not just those available to the calling user. This allows the client application to display a layout appropriate for a given user profile. For example, suppose User A owns a record, and this record has record type X set. If User B tries to view this record, then the client application can display the record using the layout associated with this record type for User B's profile (even if the record type is not available for the user).
recordTypeSelectorRequired	boolean	If true, a record type selector page is required; if false, use the default record type.

DescribeLayout

Represents a specific layout for the specified `sObjectType`. Each `DescribeLayout` is referenced by its unique layout ID and consists of two types of views (represented in this object as arrays of `DescribeLayoutSection`):

- **Detail view**—Read-only display of the object. In a detail layout, certain pieces of information (such as address details) might be aggregated into a single `DescribeLayoutItem`.
- **Edit view**—Editable display of the object. In an edit layout, individual pieces of information (such as an address) will be broken up into separate fields.

An individual `DescribeLayout` consists of these fields:

Name	Type	Description
buttonLayoutSection	DescribeLayoutButtonSection	Standard and custom button sections associated with the specified layout.
detailLayoutSections	DescribeLayoutSection[]	Layout section(s) for the detail view.
editLayoutSections	DescribeLayoutSection[]	Layout section(s) for the edit view.
highlightsPanelLayoutSection	DescribeLayoutSection[]	Layout section(s) for the highlights panel view.
multirowEditLayoutSections	DescribeLayoutSection[]	Layout section(s) for the multiline layout view. This field is available in API version 35.0 and later.
id	ID	Unique ID of this layout. For information on IDs, see ID Field Type .
quickActionList	DescribeQuickActionListResult	List of actions associated with the specified layout. This field is available in API version 28.0 and later.
relatedContent	RelatedContent	Mobile Cards section associated with the specified layout. This field is available in API version 29.0 and later.

Name	Type	Description
relatedLists	RelatedList[]	Related list(s) associated with the specified layout.

DescribeLayoutButtonSection

Represents one of two sections of the layout containing either standard or custom buttons.

Name	Type	Description
detailButtons	DescribeLayoutButton[]	Standard or custom button(s) associated with the specified button section.

DescribeLayoutButton

Represents a single standard button, custom button, or custom link in a [DescribeLayout](#).

Name	Type	Description
behavior	WebLinkWindowType	What the button or link does when clicked, such as execute JavaScript or open its content source in a new window, for example. This field is available in API version 31.0 and later.
colors	DescribeColor[]	Array of color information for icons associated with this button or link. Each color is associated with a theme. This field is available in API version 32.0 and later.
content	string	The API name of the Visualforce page or s-control being delivered. This field is available in API version 31.0 and later.
contentSource	WebLinkType	The content source of the custom button or link. The <code>contentSource</code> for a standard button which hasn't been overridden is <code>null</code> . This field is available in API version 31.0 and later.
custom	boolean	Required. Indicates whether this is a custom button or link (<code>true</code>) or not (<code>false</code>).
encoding	string	Type of encoding assigned to the URL called by the button or link. Valid values are: <ul style="list-style-type: none"> • <code>UTF-8</code>—Unicode (UTF-8) • <code>ISO-8859-1</code>—General US & Western Europe (ISO-8859-1, ISO-LATIN-1) • <code>Shift_JIS</code>—Japanese (Shift-JIS) • <code>ISO-2022-JP</code>—Japanese (JIS) • <code>EUC-JP</code>—Japanese (EUC-JP) • <code>x-SJIS_0213</code>—Japanese (Shift-JIS_2004)

Name	Type	Description
		<ul style="list-style-type: none"> • <code>ks_c_5601-1987</code>—Korean (<code>ks_c_5601-1987</code>) • <code>Big5</code>—Traditional Chinese (Big5) • <code>GB2312</code>—Simplified Chinese (GB2312) • <code>Big5-HKSCS</code>—Traditional Chinese Hong Kong (Big5-HKSCS) <p>This field is available in API version 31.0 and later.</p>
<code>height</code>	<code>int</code>	<p>The height (in pixels) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>, <code>sidebar</code>, or <code>noSidebar</code>.</p> <p>This field is available in API version 31.0 and later.</p>
<code>icons</code>	<code>Describelcon[]</code>	<p>Array of icons for this button or link. Each icon is associated with a theme. This field is available in API version 29.0 and later.</p>
<code>label</code>	<code>string</code>	<p>Label for the button or link displayed in the Salesforce user interface.</p>
<code>menubar</code>	<code>boolean</code>	<p>Indicates whether the menu bar displays (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>.</p> <p>This field is available in API version 31.0 and later.</p>
<code>name</code>	<code>string</code>	<p>API name of the button or link.</p>
<code>overridden</code>	<code>boolean</code>	<p>Required. Indicates whether a standard button has been overridden (<code>true</code>) or not (<code>false</code>).</p> <p>This field is available in API version 31.0 and later.</p>
<code>resizeable</code>	<code>boolean</code>	<p>Indicates whether the new window is resizable (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>.</p> <p>This field is available in API version 31.0 and later.</p>
<code>scrollbars</code>	<code>boolean</code>	<p>Indicates whether scrollbars display (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>.</p> <p>This field is available in API version 31.0 and later.</p>
<code>showsLocation</code>	<code>boolean</code>	<p>Indicates whether the address bar displays (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>.</p> <p>This field is available in API version 31.0 and later.</p>
<code>showsStatus</code>	<code>boolean</code>	<p>Indicates whether the status bar displays (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code>.</p> <p>This field is available in API version 31.0 and later.</p>

Name	Type	Description
toolbar	boolean	Indicates whether the toolbars display (<code>true</code>) or not (<code>false</code>) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code> . This field is available in API version 31.0 and later.
url	string	The URL called by the button or link. This field is <code>null</code> for standard buttons in a related list. This field is available in API version 31.0 and later.
width	int	The width (in pixels) when a button or link's <code>behavior</code> field value is set to <code>newWindow</code> . This field is available in API version 31.0 and later.
windowPosition	WebLinkPosition	Indicates the window position when a button or link's <code>behavior</code> field value is set to <code>newWindow</code> . This field is available in API version 31.0 and later.

DescribeLayoutComponent

Represents the smallest unit in a layout—a field or a separator. To reference a field for display, a client application uses the following notation to reference a field in the `describeSObjects()` call: `LayoutComponent.fieldName`.

In API version 31.0 and later, `DescribeLayoutComponent` is extended with `FieldLayoutComponent` if both the `LayoutComponentType` value is `Field`, and the field being described is either the compound field `Address` or the compound field `Person Name`.

Name	Type	Description
displayLines	int	The number of vertical lines displayed for a field in the edit view. Applies to <code>textarea</code> and multi-select picklist fields.
tabOrder	int	Indicates the tab order for the item in the row.
type	LayoutComponentType	The <code>LayoutComponentType</code> for this <code>LayoutComponent</code> .
value	string	Value of this <code>LayoutComponent</code> . The name of the field if the <code>LayoutComponentType</code> value is <code>Field</code> . The API name of the canvas app if the <code>LayoutComponentType</code> value is <code>Canvas</code> .

DescribeLayoutFeedFilter

Represents an individual feed filter option that you can use to filter the feed.

Name	Type	Description
label	string	The label of the filter.
name	string	The API name of the filter.

Name	Type	Description
type	FeedLayoutFilterType enum	Standard feed filter types: <ul style="list-style-type: none">• AllUpdates• FeedItemType

DescribeLayoutFeedView

Represents the layout of the feed view for a feed-based page layout.

Name	Type	Description
feedFilters	DescribeLayoutFeedFilter[]	Lists the feed filter options that are displayed with the feed.

DescribeLayoutItem

Represents an individual item in a [DescribeLayoutRow](#). A DescribeLayoutItem consists of a set of components ([DescribeLayoutComponent](#)), each of which is either a field or a separator. For most fields on a layout, there is only one component per layout item. However, in a display-only view, the DescribeLayoutItem might be a composite of the individual fields (for example, an address can consist of street, city, state, country, and postal code data). On the corresponding edit view, each component of the address field would be split up into separate DescribeLayoutItems.

Name	Type	Description
editable	boolean	Indicates whether this DescribeLayoutItem can be edited (<code>true</code>) or not (<code>false</code>). This field is available in API version 30.0 and below. It was replaced by the <code>editableForNew</code> and <code>editableForUpdate</code> fields in API version 31.0.
editableForNew	boolean	Indicates whether a new DescribeLayoutItem can be edited when creating a new record (<code>true</code>) or not (<code>false</code>). This field is available in API version 31.0 and later.
editableForUpdate	boolean	Indicates whether an existing DescribeLayoutItem can be edited when editing a record (<code>true</code>) or not (<code>false</code>). This field is available in API version 31.0 and later.
label	string	Label text for this DescribeLayoutItem.
layoutComponents	DescribeLayoutComponent[]	DescribeLayoutComponent for this DescribeLayoutItem.
placeholder	boolean	Indicates whether this DescribeLayoutItem is a placeholder (<code>true</code>) or not (<code>false</code>). If <code>true</code> , then this DescribeLayoutItem is blank.
required	boolean	Indicates whether this DescribeLayoutItem is required (<code>true</code>) or not (<code>false</code>). This is useful to know if, for example, you wanted to render required fields in a contrasting color (such as red).

DescribeLayoutRow

Represents a row in a [DescribeLayoutSection](#). A [DescribeLayoutRow](#) consists of one or more [DescribeLayoutItem](#) objects. For each [DescribeLayoutRow](#), a [DescribeLayoutItem](#) refers either to a specific field or to an “empty” [DescribeLayoutItem](#) (a [DescribeLayoutItem](#) that contains no [DescribeLayoutComponent](#) objects). An empty [DescribeLayoutItem](#) can be returned when a given [DescribeLayoutRow](#) is sparse (for example, containing more fields on the right column than on the left column). Where there are gaps in the layout, an empty [DescribeLayoutItem](#) is returned as a placeholder.

Name	Type	Description
layoutItems	DescribeLayoutItem []	Refers to either a specific field or to an empty LayoutItem (a LayoutItem that contains no DescribeLayoutComponent objects).
numItems	int	Number of layoutItems . This information is redundant but, due to a bug in a popular SOAP toolkit, was required to avoid serialization problems.

DescribeLayoutSection

Represents a section of a [DescribeLayout](#) and consists of one or more columns and one or more rows (an array of [DescribeLayoutRow](#)).

Name	Type	Description
columns	int	Number of columns in this DescribeLayoutSection .
heading	string	Heading text (label) for this DescribeLayoutSection .
layoutRows	DescribeLayoutRow []	Array of one or more DescribeLayoutRow objects.
parentLayoutId	ID	The ID of the layout upon which this DescribeLayoutSection resides. This field is available in API version 35.0 and later.
rows	int	Number of rows in this DescribeLayoutSection .
tabOrder	string	Indicates the tab order for the fields in the section in the edit view. Valid values are: <ul style="list-style-type: none">• LeftToRight• TopToBottom This field is available in API version 31.0 and later.
useCollapsibleSection	boolean	Indicates whether this DescribeLayoutSection is a collapsible section, also known as a “twistie” (<code>true</code>), or not (<code>false</code>).
useHeading	boolean	Indicates whether to display the heading (<code>true</code>) or not (<code>false</code>).

DescribeQuickActionListResult

Represents a list of actions assigned to the page layout. Available in API version 28.0 and later.

Name	Type	Description
quickActionListItems	DescribeQuickActionListItemResult[]	Array of zero or more QuickActionListItemResult objects.

DescribeQuickActionListItemResult

Represents a QuickAction assigned to the actions list for a page layout. Available in API version 28.0 and later.

Name	Type	Description
colors	DescribeColor[]	Array of color information. Each color is associated with a theme. This field is available in API version 29.0 and later.
iconUrl	string	The URL of the icon associated with the action. This icon URL corresponds to the 32x32 icon used for the current Salesforce theme, introduced in Spring '10.
icons	Describelcon[]	Array of icons for this action. Each icon is associated with a theme. This field is available in API version 29.0 and later.
label	string	The label of the action.
miniIconUrl	string	The URL of the mini icon associated with the action. This icon URL corresponds to the 16x16 icon used for the current Salesforce theme, introduced in Spring '10.
quickActionName	string	The API name of the action.
targetObjectType	string	The API name of the action's target object.
type	string	The QuickActionType of the action. Valid values are: <ul style="list-style-type: none"> • Create • VisualforcePage

CustomLinkComponent

When the [LayoutComponentType](#) value is `CustomLink`, this type contains information about a single custom link on the page layout.

Name	Type	Description
customLink	DescribeLayoutButton	A single LayoutComponent object of type <code>CustomLink</code> .

FieldLayoutComponent

Extends the information returned by [DescribeLayoutComponent](#). When the [LayoutComponentType](#) value is `Field`, and the field being described is an `Address` or `Person` Name field, FieldLayoutComponent includes information about the field's components.

Available in API version 31.0 and later.

Name	Type	Description
components	DescribeLayoutComponent[]	Array of zero or more LayoutComponent objects of type Field.
fieldType	FieldType	The field type.

Sample Code for Usage of FieldLayoutComponent

```
DescribeLayoutComponent layoutComponent = layoutComponents[n];
// Look for a component representing the BillingAddress field
if (layoutComponent.getType() == LayoutComponentType.Field.toString() &&
    layoutComponent.getValue().equals("BillingAddress")) {
    // Cast this component as a FieldLayoutComponent
    DescribeLayoutComponent.FieldLayoutComponent addressFieldComponent =
(FieldLayoutComponent)layoutComponent;
    // At this point you can access addressFieldComponent
    FieldLayoutComponent-specific methods such as getComponents() or
    getFieldtype()
}
```

LayoutComponentType

Represents the type for a [DescribeLayoutComponent](#). Contains one of these values:

- AnalyticsCloud—An Wave Analytics dashboard on the page layout. Available in API version 34.0 and later.
- Canvas—A canvas component on the page layout. This layout component type is available in API version 31.0 and later.
- CustomLink—A custom link on the page layout.
- EmptySpace—A blank space on the page layout.
- ExpandedLookup—An Expanded Lookup component in the Mobile Cards section of the page layout.
- Field—Field name. A mapping to the `name` field on the [describeSObjectResult](#).
- ReportChart—A report chart on the page layout.
- SControl—Reserved for future use.
- Separator—Separator character, such as a semicolon (:) or slash (/).
- visualforcePage—A Visualforce component on the page layout.

PicklistForRecordType

Represents a single record type picklist in a [RecordTypeMapping](#). The `picklistName` matches up with the `name` attribute of each field in the `fields` array in [describeSObjectResult](#). The `picklistValues` are the set of acceptable values for the `recordType`.

Name	Type	Description
picklistName	string	Name of the picklist.

Name	Type	Description
picklistValues	PicklistEntry[]	<p>Set of picklist values associated with the <code>recordTypeIds</code> in the <code>RecordTypeMapping</code>.</p> <p>Note: If you retrieve <code>picklistValues</code>, the <code>validFor</code> value is null. If you need the <code>validFor</code> value, get it from the <code>PicklistEntry</code> object obtained from the <code>Field</code> object associated with the <code>DescribeSObjectResult</code>.</p>

RecordTypeMapping

Represents a single record type mapping in the `recordTypeMappings` field in a `DescribeLayoutResult` object. This object is a map of valid `recordTypeIds` to `layoutId`. For displaying a detail view, a client application uses this mapping to determine which layout is associated with the record type on the record. For displaying an edit view, a client application uses this mapping to determine which layout to use (and possibly to allow the user to choose between multiple record types); it will also determine the set of available picklist values.

Name	Type	Description
available	boolean	Indicates whether this record type is available (<code>true</code>) or not (<code>false</code>). Availability is used to display a list of available record types to the user when they are creating a new record.
defaultRecordTypeMapping	boolean	Indicates whether this is the default record type mapping (<code>true</code>) or not (<code>false</code>).
layoutId	ID	ID of the layout associated with this record type.
name	string	Name of this record type.
picklistsForRecordType	PicklistForRecordType[]	Record type picklist(s) mapped to the <code>recordTypeIds</code> .
recordTypeId	ID	ID of this record type.

 **Note:** Some fields previously in this result have moved to `RecordTypeInfo`.

RelatedContent

Represents the Mobile Cards section in a `DescribeLayout`. Available in API version 29.0 and later.

Name	Type	Description
relatedContentItems	DescribeRelatedContentItem[]	An array of items in the Mobile Cards section of the page layout.

DescribeRelatedContentItem

Represents an individual item in the `DescribeRelatedContentItem` list. Available in API version 29.0 and later.

Name	Type	Description
describeLayoutItem	DescribeLayoutItem	An individual layout item in the Mobile Cards section. Must be wrapped in a DescribeRelatedContentItem to be added to the Mobile Cards section.

RelatedList

Represents a single related list in a [DescribeLayoutResult](#).

Name	Type	Description
buttons	DescribeLayoutButton[]	Buttons associated with this related list. This field is available in API version 32.0 and later.
columns	RelatedListColumn[]	<p>Columns associated with this related list.</p> <p>You can pair this value with Field to achieve a number of useful tasks, including determining whether the field is:</p> <ul style="list-style-type: none"> • A name field, in order to present a link to the detail • Sortable, (to allow the user to include it in an ORDER BY clause to sort the rows by the given column) • A currency field, to include the currency symbol or code
custom	boolean	If true, this related list is custom.
field	string	Name of the field on the related (associated) object that establishes the relationship with the associating object. For example, for the Contact related list on Account , the value is <code>AccountId</code> .
label	string	Label for the related list, displayed in the Salesforce user interface.
limitRows	int	Number of rows to display.
name	string	Name of the ChildRelationship in the DescribeSObjectResult for the <code>sObjectType</code> which was provided as the argument to DescribeLayout .
sobject	string	Name of the <code>sObjectType</code> that is the row type for rows within this related list.
sort	RelatedListSort[]	If not null, the column(s) that should be used to order the related objects.

RelatedListColumn

Represents a single field in a related list returned by [DescribeLayoutResult](#).

Name	Type	Description
field	string	API name of the field. This value is always of the form object_type.field_name . For example, if <code>name</code> is <code>Contact.Account.Owner.Alias</code> , then this value is <code>User.Alias</code> .
fieldApiName	string	SOQL field syntax for the field in relation to the main <code>sObject</code> for the related list. This value is always of the form object_type.field_name . Unlike <code>name</code> , it doesn't return a value in the Translating Results format.
format	string	Display in date or dateTime format.
label	string	Label of the field.
lookupId	string	Optional SOQL field syntax to retrieve the lookup ID value for the main related list <code>sObject</code> . This value may be an expression that uses SOQL relationship query dot notation. For example, if the related list <code>sObjectType</code> is <code>Case</code> and the column display value is <code>Owner.Alias</code> , then the lookup ID value would be <code>Owner.Id</code> .
name	string	SOQL field syntax for the field in relation to the main <code>sObject</code> for the related list. This value may be an expression that uses SOQL relationship query dot notation, or it may use the Translating Results or <code>convertCurrency()</code> format. For example, if the related list <code>sObjectType</code> is <code>Case</code> , then the value might be <code>Owner.Alias</code> or it might be <code>toLabel(Case.Status)</code> .

RelatedListSort

Represents the sorting preference for objects in the related list.

Name	Type	Description
column	string	Name of the field that is used to order the related objects.
ascending	boolean	If <code>true</code> , sort order is ascending. If <code>false</code> , descending.

Although in most cases there is only one RelatedListSort in the array, for some special standard related lists, there is more than one. If there is more than one, the RelatedListSorts are ordered according to how they should be included in a corresponding SOQL query, for example:

```
ORDER BY relatedListSort[0].getColumn() DIRECTION, relatedListSort[1].getColumn() DIRECTION
```

WebLinkPosition

Represents the window position for a new window opened upon clicking a `DescribeLayoutButton`. Applies only to custom buttons. Available in API version 31.0 and later. Contains one of these values:

- `fullScreen`—The new window opens in a full screen. If this option is selected, any width or height parameters set for the new window are ignored.
- `none`—No window position preference is set.

- `topLeft`—The new window opens, positioned at the top left of the screen.

WebLinkType

Represents the content being delivered by the custom button. Contains one of these values:

- `javascript`
- `page`—Visualforce page
- `sControl`
- `url`

WebLinkWindowType

Represents the behavior for a `DescribeLayoutButton`. Applies only to custom buttons. Available in API version 31.0 and later. Contains one of these values:

- `newWindow`—The custom button's content opens in a new browser window.
- `noSidebar`—The custom button's content displays in the existing browser window without a sidebar.
- `onClickJavaScript`—Valid only when the `DescribeLayoutButton`'s `contentSource` field value is `javascript`. Clicking the button or link executes JavaScript.
- `replace`—The custom button's content displays in the existing browser window without a sidebar or header.
- `sidebar`—The custom button's content displays in the existing browser window with a sidebar.

describePrimaryCompactLayouts ()

Retrieves metadata about the primary compact layout for each of the specified object types. Information returned is limited to 100 objects.

Syntax

```
DescribeCompactLayout[] primaryCompactLayouts =
connection.describePrimaryCompactLayouts(string[] sObjectType)
```

Usage

Use this call to retrieve information about the primary compact layout for the given object types. This call returns metadata about a given primary compact layout. For more information about compact layouts, see the Salesforce Help.

Sample Code—Java

```
public void testDescribePrimaryCompactLayoutsSample() {
    try {
        String[] objectsToDescribe = new String[] {"Account", "Lead"};
        DescribeCompactLayout[] primaryCompactLayouts =
connection.describePrimaryCompactLayouts(objectsToDescribe);

        for (int i = 0; i < primaryCompactLayouts.length; i++) {
```

```

DescribeCompactLayout cLayout = primaryCompactLayouts[i];
System.out.println(" There is a compact layout with name: " + cLayout.getName());

// Write the objectType
System.out.println(" This compact layout is the primary compact layout for: " +
cLayout.getObjectType();

DescribeLayoutItem[] fieldItems = cLayout.getFieldItems();
System.out.println(" There are " + fieldItems.length + " fields in this compact
layout.");

// Write field items
for (int j = 0; j < fieldItems.length; j++) {
    System.out.println(j + " This compact layout has a field with name: " +
fieldItems[j].getLabel());
}

DescribeLayoutItem[] imageItems = cLayout.getImageItems();
System.out.println(" There are " + imageItems.length + " image fields in this
compact layout.");

// Write the image items
for (int j = 0; j < imageItems.length; j++) {
    System.out.println(j + " This compact layout has an image field with name: " +
imageItems[j].getLabel());
}

DescribeLayoutButton[] actions = cLayout.getActions();
System.out.println(" There are " + actions.length + " buttons in this compact
layout.");

// Write the action buttons
for (int j = 0; j < actions.length; j++) {
    System.out.println(j + " This compact layout has a button with name: " +
actions[j].getLabel());
}

} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
}

```

Arguments

Name	Type	Description
sObjectTypes	String[]	An array of one or more objects. The specified values must be valid objects for your organization.

Response

[DescribeCompactLayout](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

describeQuickActions()

Retrieves details about specified actions.

Syntax

```
DescribeQuickActionResult[] = connection.describeQuickActions(string[] quickActionNames);
```

Usage

Use the `describeQuickActions()` call to retrieve details for specified actions. In API version 28.0, the `describeQuickActions()` call takes the action name in the form of `ParentEntity.ActionName`. In API version 29.0 and greater, it takes the action name in the form of `ContextEntity.ActionName`. Returns an array of `DescribeQuickActionResult`. You might first call [describeAvailableQuickActions\(\)](#) for a list of actions available for a specified context and then use `describeQuickActions()` to obtain details about specific actions.

Sample—Java

This sample retrieves and displays publisher action details for a create action on the Account object.

```
public void example() throws Exception {
    DescribeQuickActionResult[] result =
        conn.describeQuickActions(new String[]
            { "Account.QuickCreateContact", "Account.QuickCreateTask" });
    for(DescribeQuickActionResult r : result) {
        assert r != null;
        DescribeQuickActionDefaultValue [] describeQuickActionDefaultValues =
r.getDefaultValues();
        for(DescribeQuickActionDefaultValue defaultValue : describeQuickActionDefaultValues)
{
            System.out.println("Target Object Field: " + defaultValue.getField());
            System.out.println("Target Object Field's default Value: " +
defaultValue.getDefaultValue());
        }
        System.out.println("Action name: " + r.getName());
        System.out.println("Action label: " + r.getLabel());
        System.out.println("ParentOrContext object: " + r.getSourceSObjectType());
        System.out.println("Target object: " + r.getTargetSObjectType());
        System.out.println("Target object record type: " + r.getTargetRecordTypeId());
    }
}
```

```

        System.out.println("Relationship field: " + r.getTargetParentField());
        System.out.println("Quick action type: " + r.getType());
        System.out.println("VF page name for custom actions: " +
r.getVisualforcePageName());
        System.out.println("Icon name: " + r.getIconName());
        System.out.println("Icon URL: " + r.getIconUrl());
        System.out.println("Mini icon URL: " + r.getMiniIconUrl());
        assert r.getLayout() != null;
        System.out.println("Height of VF page for custom actions: " + r.getHeight());
        System.out.println("Width of VF page for custom actions: " + r.getWidth());
    }
}

```

Arguments

Name	Type	Description
quickActions	string[]	An array of quick actions to be retrieved.

Response

[DescribeQuickActionResult](#)

DescribeQuickActionResult

The `describeQuickActions()` call returns an array of `DescribeQuickActionResult` objects. Each `DescribeQuickActionResult` object represents a quick action for a specified object.

Name	Type	Description
actionEnumOrId	string	The unique ID for the action. If the action doesn't have an ID, its API name is used. This field is available in API version 35.0 and later.
canvasApplicationName	string	The name of your Force.com Canvas application, if you use it.
colors	DescribeColor[]	Array of color information. Each color is associated with a theme. This field is available in API version 29.0 and later.
defaultValues	DescribeQuickActionDefaultValue[]	The action's default values.
height	int	The height in pixels of the action pane.
iconName	string	Name of icon used for the action. If a custom icon is not used, this value will not be set.

Name	Type	Description
iconUrl	string	URL of icon used for the action. This icon URL corresponds to the 32x32 icon used for the current Salesforce theme, introduced in Spring '10, or the custom icon, if there is one.
icons	Describelcon[]	<p>Array of icons. Each icon is associated with a theme.</p> <p>If no custom icon was associated with the quick action and the quick action creates a specific object, the icons will correspond to the icons used for the created object. For example, if the quick action creates an Account, the icon array will contain the icons used for Account.</p> <p>If a custom icon was associated with the quick action, the array will contain that custom icon.</p> <p>This field is available in API version 29.0 and later. API version 32.0 and later returns different icons than in earlier API versions.</p>
label	string	Label of the action.
layout	DescribeLayoutSection	The section of the layout where the action resides.
lightningComponentBundleId	ID	<p>If <code>type</code> is <code>LightningComponent</code>, the ID of the Lightning component bundle called by the action.</p> <p>This field is available in API version 38.0 and later.</p>
lightningComponentBundleName	string	<p>If <code>type</code> is <code>LightningComponent</code>, the name of the Lightning component bundle called by the action.</p> <p>This field is available in API version 38.0 and later.</p>
miniIconUrl	string	The icon's URL. This icon URL corresponds to the 16x16 icon used for the current Salesforce theme, introduced in Spring '10, or the custom icon, if there is one.
name	string	Name of the action.
contextSobjectType	string	The object used for the action. Named <code>sourceSobjectType</code> in version 29.0 and earlier.
showQuickActionVfHeader	boolean	Whether or not the Visualforce quick action header and footer should be shown. If set to <code>false</code> , then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren't displayed.
targetParentField	string	The parent object type of the action. Links the target object to the parent object. For example, use Account if the target object is Contact and the parent object is Account.
targetRecordTypeID	ID	The record type of the target record.
targetSobjectType	string	The action's target object type.

Name	Type	Description
type	string	The action's type. Valid values are: <ul style="list-style-type: none">• Canvas• Create• LightningComponent (This value is available in API version 38.0 and later.)• LogACall• Post• SendEmail (This value is available in API version 31.0 and later.)• SocialPost• Update• VisualforcePage
visualforcePageName	string	If <code>type</code> is Visualforce, the page name of the associated page for the action.
visualforcePageUrl	string	If <code>type</code> is Visualforce, the URL of the associated page for the action.
width	int	If a custom action is created, this is the width in pixels of the action pane.

DescribeQuickActionDefaultValue

Represents the default values of fields to use in default layouts.

Name	Type	Description
defaultValue	string	The value of the auto-populated default action.
field	string	The field name of the action.

DescribeLayoutSection

Represents a section of a [DescribeLayout](#) and consists of one or more columns and one or more rows (an array of [DescribeLayoutRow](#)).

Name	Type	Description
columns	int	Number of columns in this DescribeLayoutSection .
heading	string	Heading text (label) for this DescribeLayoutSection .
layoutRows	DescribeLayoutRow []	Array of one or more DescribeLayoutRow objects.
parentLayoutId	ID	The ID of the layout upon which this DescribeLayoutSection resides. This field is available in API version 35.0 and later.

Name	Type	Description
rows	int	Number of rows in this DescribeLayoutSection .
tabOrder	string	Indicates the tab order for the fields in the section in the edit view. Valid values are: <ul style="list-style-type: none"> • LeftToRight • TopToBottom This field is available in API version 31.0 and later.
useCollapsibleSection	boolean	Indicates whether this DescribeLayoutSection is a collapsible section, also known as a "twistie" (<code>true</code>), or not (<code>false</code>).
useHeading	boolean	Indicates whether to display the heading (<code>true</code>) or not (<code>false</code>).

DescribeLayoutRow

Represents a row in a [DescribeLayoutSection](#). A DescribeLayoutRow consists of one or more [DescribeLayoutItem](#) objects. For each DescribeLayoutRow, a [DescribeLayoutItem](#) refers either to a specific field or to an "empty" [DescribeLayoutItem](#) (a [DescribeLayoutItem](#) that contains no [DescribeLayoutComponent](#) objects). An empty [DescribeLayoutItem](#) can be returned when a given DescribeLayoutRow is sparse (for example, containing more fields on the right column than on the left column). Where there are gaps in the layout, an empty [DescribeLayoutItem](#) is returned as a placeholder.

Name	Type	Description
layoutItems	DescribeLayoutItem []	Refers to either a specific field or to an empty LayoutItem (a LayoutItem that contains no DescribeLayoutComponent objects).
numItems	int	Number of <code>layoutItems</code> . This information is redundant but, due to a bug in a popular SOAP toolkit, was required to avoid serialization problems.

DescribeLayoutItem

Represents an individual item in a [DescribeLayoutRow](#). A DescribeLayoutItem consists of a set of components ([DescribeLayoutComponent](#)), each of which is either a field or a separator. For most fields on a layout, there is only one component per layout item. However, in a display-only view, the DescribeLayoutItem might be a composite of the individual fields (for example, an address can consist of street, city, state, country, and postal code data). On the corresponding edit view, each component of the address field would be split up into separate DescribeLayoutItems.

Name	Type	Description
editable	boolean	Indicates whether this DescribeLayoutItem can be edited (<code>true</code>) or not (<code>false</code>). This field is available in API version 30.0 and below. It was replaced by the <code>editableForNew</code> and <code>editableForUpdate</code> fields in API version 31.0.

Name	Type	Description
editableForNew	boolean	Indicates whether a new DescribeLayoutItem can be edited when creating a new record (<code>true</code>) or not (<code>false</code>). This field is available in API version 31.0 and later.
editableForUpdate	boolean	Indicates whether an existing DescribeLayoutItem can be edited when editing a record (<code>true</code>) or not (<code>false</code>). This field is available in API version 31.0 and later.
label	string	Label text for this DescribeLayoutItem.
layoutComponents	DescribeLayoutComponent[]	DescribeLayoutComponent for this DescribeLayoutItem.
placeholder	boolean	Indicates whether this DescribeLayoutItem is a placeholder (<code>true</code>) or not (<code>false</code>). If <code>true</code> , then this DescribeLayoutItem is blank.
required	boolean	Indicates whether this DescribeLayoutItem is required (<code>true</code>) or not (<code>false</code>). This is useful to know if, for example, you wanted to render required fields in a contrasting color (such as red).

DescribeLayoutComponent

Represents the smallest unit in a layout—a field or a separator. To reference a field for display, a client application uses the following notation to reference a field in the `describeSObjects()` call: `LayoutComponent.fieldName`.

Name	Type	Description
displayLines	int	The number of vertical lines displayed for a field in the edit view. Applies to <code>textarea</code> and multi-select picklist fields.
tabOrder	int	Indicates the tab order for the item in the row.
type	LayoutComponentType	The LayoutComponentType for this LayoutComponent.
value	string	Value of this LayoutComponent. The name of the field if the LayoutComponentType value is <code>Field</code> . The API name of the canvas app if the LayoutComponentType value is <code>Canvas</code> .

LayoutComponentType

Represents the type for a `DescribeLayoutComponent`. Contains one of these values:

- `AnalyticsCloud`—An Wave Analytics dashboard on the page layout. Available in API version 34.0 and later.
- `Canvas`—A canvas component on the page layout. This layout component type is available in API version 31.0 and later.
- `CustomLink`—A custom link on the page layout.
- `EmptySpace`—A blank space on the page layout.
- `ExpandedLookup`—An Expanded Lookup component in the Mobile Cards section of the page layout.
- `Field`—Field name. A mapping to the `name` field on the `describeSObjectResult`.

- ReportChart—A report chart on the page layout.
- SControl—Reserved for future use.
- Separator—Separator character, such as a semicolon (:) or slash (/).
- VisualforcePage—A Visualforce component on the page layout.

describeSearchScopeOrder ()

Retrieves an ordered list of the objects in a user's default global search scope.

Syntax

```
DescribeSearchScopeOrderResult[] describeSearchScopeOrderResults =
connection.describeSearchScopeOrder();
```

Usage

Use `describeSearchScopeOrder ()` to retrieve an ordered list of objects in the default global search scope of a logged-in user. Global search keeps track of which objects the user interacts with and how often and arranges the search results accordingly. Objects used most frequently appear at the top of the list. The returned list reflects the object order in the user's default search scope, including any pinned objects on the user's search results page. This call is useful if you want to implement a custom search results page using the optimized global search scope.



Note: You must enable Chatter to enable global search.

Sample Code—Java

This sample shows how to retrieve the global search scope for a user and then iteratively display the name of each object in the scope.

```
public void describeSearchScopeOrderSample() {
    try {
        //Get the order of objects in search smart scope for the logged-in user
        DescribeSearchScopeOrderResult[] describeSearchScopeOrderResults =
            connection.describeSearchScopeOrder();
        //Iterate through the results and display the name of each object
        for (int i = 0; i < describeSearchScopeOrderResults.length; i++) {
            System.out.println(describeSearchScopeOrderResults[i].getName());
        }
    }
    catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Arguments

None.

Response

An array of [DescribeSearchScopeOrderResult](#) objects

Fault

[UnexpectedErrorFault](#)

SEE ALSO:

[API Call Basics](#)

DescribeSearchScopeOrderResult

The `describeSearchScopeOrder()` call returns an array of `DescribeSearchScopeOrderResult` objects. Each `DescribeSearchScopeOrderResult` object represents an object in the user's global search scope. The list reflects the order of the objects in the user's scope, including any pinned objects. The `DescribeSearchScopeOrderResult` object has the following properties.

Name	Type	Description
keyPrefix	string	Three-character prefix code in the object ID. Object IDs are prefixed with three-character codes that specify the type of the object. For example, Account objects have a prefix of 001 and Opportunity objects have a prefix of 006. Note that a key prefix can sometimes be shared by multiple objects so it does not always uniquely identify an object.
name	string	Name of the object. English only.

describeSearchLayouts ()

Retrieves the search result layout configuration for one or more objects.

Syntax

```
DescribeSearchLayoutResult[] = binding.describeSearchLayouts(string sObjectType[]);
```

Usage

Use `describeSearchLayouts()` to retrieve search layout information for one or more objects. This is handy when you want to create a custom search results page with the same layout settings as in Salesforce.

Sample

This sample shows how to retrieve the search result layout information for a list of objects.

```
public void describeSearchLayoutSample(String[] sObjectTypes) {
    try {
        // Get the search layout of Account and Group
        DescribeSearchLayoutResult[] searchLayoutResults =
connection.describeSearchLayouts(sObjectTypes);
        // Iterate through the results and display the label of each column
        for (int i = 0; i < sObjectTypes.length; i += 1) {
            String sObjectType = sObjectTypes[i];
            DescribeSearchLayoutResult result = searchLayoutResults[i];
            System.out.println("Top label for search results for " + sObjectType + " is " + result.getLabel() + " and should display " + result.getLimitRows() + " rows");
            System.out.println("Column labels for search results for " + sObjectType + " are: ");
            for (DescribeColumn column : result.getSearchColumns()) {
                System.out.println(column.getLabel());
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Arguments

Name	Type	Description
sObjectType	String[]	The list of objects you want to obtain search result layout configuration for. For example, if the object is a person account, specify Account, or if it is a person contact, specify Contact. The specified values must be valid objects in your organization. For a complete list of standard objects, see Standard Objects .

Response

[DescribeSearchLayoutResult](#)

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

DescribeSearchLayoutResult

The [describeSearchLayouts \(\)](#) on page 2292 call returns an array of `DescribeSearchLayoutResult` objects. Each `DescribeSearchLayoutResult` object represents the search layout configuration for each object queried for. The `DescribeSearchLayoutResult` object has the following properties.

Name	Type	Description
label	string	The browser title used for the search results page.
limitRows	int	The maximum number of rows to be displayed in the first page of search results. This number can be changed by the administrator.
searchColumns	DescribeColumn on page 2294[]	The columns associated with the search results for this object.

DescribeColumn

Represents the columns in the search layout configuration for each `DescribeSearchLayoutResult` object returned by the [describeSearchLayouts \(\)](#) on page 2292 call.

Name	Type	Description
field	string	Field reference in relation to the object it belongs to. For example, "Lead.Phone."
format	string	Field data format. For example, "date". This value can be null.
label	string	Display text for this field in the user interface. For example, "Company Phone" or just "Phone."
name	string	Field name. Use this in your SOQL query or code. For example, "Name."

describeSObject()

Describes metadata (field list and object properties) for the specified object.

 **Note:** `describeSObjects ()` supersedes `describeSObject ()`. Use `describeSObjects ()` instead of `describeSObject ()`.

Syntax

```
DescribeSObjectResult = connection.describeSObject(string sObjectType);
```

Usage

Use [describeSObject \(\)](#) to obtain metadata for a given object. You can first call [describeGlobal \(\)](#) to retrieve a list of all objects for your organization, then iterate through the list and use [describeSObject \(\)](#) to obtain metadata about individual objects.

Your client application must be logged in with sufficient access rights to retrieve metadata about your organization's data. For more information, see [Factors that Affect Data Access](#).

Sample Code—Java

This sample calls `describeSObject()` to perform describes on the Account sObject. It retrieves some properties of the sObject describe result, such as the sObject name, label, and fields. It then iterates through the fields and gets the field properties. For picklist fields, it gets the picklist values and for reference fields, it gets the referenced object names. The sample writes the retrieved sObject and field properties to the console.

```
public void describeSObjectSample() {
    try {
        // Make the describe call
        DescribeSObjectResult describeSObjectResult =
            connection.describeSObject("Account");

        // Get sObject metadata
        if (describeSObjectResult != null) {
            System.out.println("sObject name: " +
                describeSObjectResult.getName());
            if (describeSObjectResult.isCreateable())
                System.out.println("Createable");

        // Get the fields
        Field[] fields = describeSObjectResult.getFields();
        System.out.println("Has " + fields.length + " fields");

        // Iterate through each field and gets its properties
        for (int i = 0; i < fields.length; i++) {
            Field field = fields[i];
            System.out.println("Field name: " + field.getName());
            System.out.println("Field label: " + field.getLabel());

            // If this is a picklist field, show the picklist values
            if (field.getType().equals(FieldType.picklist)) {
                PicklistEntry[] picklistValues =
                    field.getPicklistValues();
                if (picklistValues != null) {
                    System.out.println("Picklist values: ");
                    for (int j = 0; j < picklistValues.length; j++) {
                        if (picklistValues[j].getLabel() != null) {
                            System.out.println("\tItem: " +
                                picklistValues[j].getLabel());
                        }
                    }
                }
            }

            // If a reference field, show what it references
            if (field.getType().equals(FieldType.reference)) {
                System.out.println("Field references the " +
                    "following objects:");
                String[] referenceTos = field.getReferenceTo();
            }
        }
    }
}
```

```
        for (int j = 0; j < referenceTos.length; j++) {
            System.out.println("\t" + referenceTos[j]);
        }
    }
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample calls `describeSObject()` to perform describes on the Account sObject. It retrieves some properties of the sObject describe result, such as the sObject name, label, and fields. It then iterates through the fields and gets the field properties. For picklist fields, it gets the picklist values and for reference fields, it gets the referenced object names. The sample writes the retrieved sObject and field properties to the console.

```
public void describeSObjectSample() {
    try {
        // Make the describe call
        DescribeSObjectResult describeSObjectResult =
            binding.describeSObject("Account");

        // Get sObject metadata
        if (describeSObjectResult != null) {
            Console.WriteLine("sObject name: " +
                describeSObjectResult.name);
            if (describeSObjectResult.createable)
                Console.WriteLine("Createable");

            // Get the fields
            Field[] fields = describeSObjectResult.fields;
            Console.WriteLine("Has " + fields.Length + " fields");

            // Iterate through each field and gets its properties
            for (int i = 0; i < fields.Length; i++) {
                Field field = fields[i];
                Console.WriteLine("Field name: " + field.name);
                Console.WriteLine("Field label: " + field.label);

                // If this is a picklist field, show the picklist values
                if (field.type.Equals(fieldType.picklist)) {
                    PicklistEntry[] picklistValues =
                        field.picklistValues;
                    if (picklistValues != null) {
                        Console.WriteLine("Picklist values: ");
                        for (int j = 0; j < picklistValues.Length; j++) {
                            if (picklistValues[j].label != null) {
                                Console.WriteLine("\tItem: " +
                                    picklistValues[j].label);
                            }
                        }
                    }
                }
            }
        }
    }
}
```

```
        }

    }

    // If a reference field, show what it references
    if (field.type.Equals(fieldType.reference)) {
        Console.WriteLine("Field references the " +
            "following objects:");
        String[] referenceTos = field.referenceTo;
        for (int j = 0; j < referenceTos.Length; j++) {
            Console.WriteLine("\t" + referenceTos[j]);
        }
    }
}

} catch (SoapException e) {
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
```

Arguments

Name	Type	Description
sObjectType	string	Object. The specified value must be a valid object for your organization. For a complete list of objects, see Standard Objects .

Response

DescribeSObjectResult

Faults

InvalidSObjectFault

UnexpectedErrorFault

SEE ALSO:

describeSObjects()

describeGlobal()

API Call Basics

Using the Partner WSDL

https://developer.salesforce.com/page/Sample_SOAP_Messages

describeSObjectResult

The `describeSObject()` call returns a `DescribeSObjectResult` object.

 **Note:** `describeSObjects()` supersedes `describeSObject()`. Use `describeSObjects()` instead of `describeSObject()`.

describeSObjects()

An array-based version of `describeSObject()`; describes metadata (field list and object properties) for the specified object or array of objects.

 **Note:** Use this call instead of `describeSObject()`.

Syntax

```
DescribeSObjectResult [] = connection.describeSObjects(string sObjectType[] );
```

Usage

Use `describeSObjects()` to obtain metadata for a given object or array of objects. You can first call `describeGlobal()` to retrieve a list of all objects for your organization, then iterate through the list and use `describeSObjects()` to obtain metadata about individual objects. The `describeSObjects()` call is limited to a maximum of 100 objects returned.

Your client application must be logged in with sufficient access rights to retrieve metadata about your organization's data. For more information, see [Factors that Affect Data Access](#).

In organizations where person accounts are enabled, this call shows [Accounts](#) as not createable if the profile does not have access to any business account record types.

Sample Code—Java

This sample calls `describeSObjects()` to perform describes on account, contact, and lead. It iterates through the sObject describe results, gets the properties and fields for each sObject in the result, and writes them to the console. For picklist fields, it writes the picklist values. For reference fields, it writes the referenced object names.

```
public void describeSObjectsSample()
{
    try {
        // Call describeSObjectResults and pass it an array with
        // the names of the objects to describe.
        DescribeSObjectResult[] describeSObjectResults =
            connection.describeSObjects(
                new String[] { "account", "contact", "lead" });

        // Iterate through the list of describe sObject results
        for (int i=0;i < describeSObjectResults.length; i++)
        {
            DescribeSObjectResult desObj = describeSObjectResults[i];
            // Get the name of the sObject
            String objectName = desObj.getName();
            System.out.println("sObject name: " + objectName);

            // For each described sObject, get the fields
        }
    }
}
```

```
Field[] fields = desObj.getFields();

// Get some other properties
if (desObj.getActivateable()) System.out.println("\tActivateable");

// Iterate through the fields to get properties for each field
for(int j=0;j < fields.length; j++)
{
    Field field = fields[j];
    System.out.println("\tField: " + field.getName());
    System.out.println("\t\tLabel: " + field.getLabel());
    if (field.isCustom())
        System.out.println("\t\tThis is a custom field.");
    System.out.println("\t\tType: " + field.getType());
    if (field.getLength() > 0)
        System.out.println("\t\tLength: " + field.getLength());
    if (field.getPrecision() > 0)
        System.out.println("\t\tPrecision: " + field.getPrecision());

    // Determine whether this is a picklist field
    if (field.getType() == FieldType.picklist)
    {
        // Determine whether there are picklist values
        PicklistEntry[] picklistValues = field.getPicklistValues();
        if (picklistValues != null && picklistValues[0] != null)
        {
            System.out.println("\t\tPicklist values = ");
            for (int k = 0; k < picklistValues.length; k++)
            {
                System.out.println("\t\t\tItem: " + picklistValues[k].getLabel());
            }
        }
    }

    // Determine whether this is a reference field
    if (field.getType() == FieldType.reference)
    {
        // Determine whether this field refers to another object
        String[] referenceTos = field.getReferenceTo();
        if (referenceTos != null && referenceTos[0] != null)
        {
            System.out.println("\t\tField references the following objects:");
            for (int k = 0; k < referenceTos.length; k++)
            {
                System.out.println("\t\t\t" + referenceTos[k]);
            }
        }
    }
}

} catch(ConnectionException ce) {
    ce.printStackTrace();
}
```

```
}
```

Sample Code—C#

This sample calls `describeSObjects()` to perform describes on account, contact, and lead. It iterates through the sObject describe results, gets the properties and fields for each sObject in the result, and writes them to the console. For picklist fields, it writes the picklist values. For reference fields, it writes the referenced object names.

```
public void describeSObjectsSample()
{
    try
    {
        // Call describeSObjectResults and pass it an array with
        // the names of the objects to describe.
        DescribeSObjectResult[] describeSObjectResults =
            binding.describeSObjects(
                new string[] { "account", "contact", "lead" });

        // Iterate through the list of describe sObject results
        foreach (DescribeSObjectResult describeSObjectResult in describeSObjectResults)
        {
            // Get the name of the sObject
            String objectName = describeSObjectResult.name;
            Console.WriteLine("sObject name: " + objectName);

            // For each described sObject, get the fields
            Field[] fields = describeSObjectResult.fields;

            // Get some other properties
            if (describeSObjectResult.activateable) Console.WriteLine("\tActivateable");

            // Iterate through the fields to get properties for each field
            foreach (Field field in fields)
            {
                Console.WriteLine("\tField: " + field.name);
                Console.WriteLine("\t\tLabel: " + field.label);
                if (field.custom)
                    Console.WriteLine("\t\tThis is a custom field.");
                Console.WriteLine("\t\tType: " + field.type);
                if (field.length > 0)
                    Console.WriteLine("\t\tLength: " + field.length);
                if (field.precision > 0)
                    Console.WriteLine("\t\tPrecision: " + field.precision);

                // Determine whether this is a picklist field
                if (field.type == fieldType.picklist)
                {
                    // Determine whether there are picklist values
                    PicklistEntry[] picklistValues = field.picklistValues;
                    if (picklistValues != null && picklistValues[0] != null)
                    {
                        Console.WriteLine("\t\tPicklist values = ");
                    }
                }
            }
        }
    }
}
```

```

        for (int j = 0; j < picklistValues.Length; j++)
        {
            Console.WriteLine("\t\t\tItem: " + picklistValues[j].label);
        }
    }

    // Determine whether this is a reference field
    if (field.type == fieldType.reference)
    {
        // Determine whether this field refers to another object
        string[] referenceTos = field.referenceTo;
        if (referenceTos != null && referenceTos[0] != null)
        {
            Console.WriteLine("\t\tField references the following objects:");
            for (int j = 0; j < referenceTos.Length; j++)
            {
                Console.WriteLine("\t\t\t" + referenceTos[j]);
            }
        }
    }
}

catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " + e.Message
        + "\n" + e.StackTrace);
}

```

Arguments

The `describeSObjects()` call takes in an array of sObjects.

Name	Type	Description
sObjectType	string	Object. The specified value must be a valid object for your organization. For a complete list of objects, see Standard Objects .

Response

DescribeSObjectResult

Faults

InvalidSObjectFault

[UnexpectedErrorFault](#)

SEE ALSO:

[describeSObject\(\)](#)
[describeGlobal\(\)](#)
[API Call Basics](#)
[Using the Partner WSDL](#)

DescribeSObjectResult

The [describeSObjects \(\)](#) call returns an array of `DescribeSObjectResult` objects. Each object has the following properties:

Name	Type	Description
<code>activateable</code>	boolean	Reserved for future use.
<code>actionOverrides</code>	ActionOverride[]	An array of action overrides. Action overrides replace the URLs specified in the <code>urlDetail</code> , <code>urlEdit</code> and <code>urlNew</code> fields. This field is available in API version 32.0 and later.
<code>childRelationships</code>	ChildRelationship[]	An array of child relationships, which is the name of the sObject that has a foreign key to the sObject being described.
<code>compactLayoutable</code>	boolean	Indicates that the object can be used in describeCompactLayouts () .
<code>createable</code>	boolean	Indicates whether the object can be created via the create () call (<code>true</code>) or not (<code>false</code>).
<code>custom</code>	boolean	Indicates whether the object is a custom object (<code>true</code>) or not (<code>false</code>).
<code>customSetting</code>	boolean	Indicates whether the object is a custom setting object (<code>true</code>) or not (<code>false</code>).
<code>deletable</code>	boolean	Indicates whether the object can be deleted via the delete () call (<code>true</code>) or not (<code>false</code>).
<code>deprecatedAndHidden</code>	boolean	Reserved for future use.
<code>feedEnabled</code>	boolean	Indicates whether Chatter feeds are enabled for the object (<code>true</code>) or not (<code>false</code>). This property is available in API version 19.0 and later.
<code>fields</code>	Field[]	Array of fields associated with the object. The mechanism for retrieving information from this list varies among development tools.
<code>keyPrefix</code>	string	Three-character prefix code in the object ID. Object IDs are prefixed with three-character codes that specify the type of the object. For example, Account objects have a prefix of 001 and Opportunity objects have a prefix of 006. Note that a key prefix can sometimes be shared by multiple objects so it does not always uniquely identify an object. Use the value of this field to determine the object type of a parent in those cases where the child may have more than one object type as parent

Name	Type	Description
		(polymorphic). For example, you may need to obtain the <code>keyPrefix</code> value for the parent of a Task or Event .
label	string	Label text for a tab or field renamed in the user interface, if applicable, or the object name, if not. For example, an organization representing a medical vertical might rename Account to <code>Patient</code> . Tabs and fields can be renamed in the Salesforce user interface. See the Salesforce online help for more information.
labelPlural	string	Label text for an object that represents the plural version of an object name, for example, "Accounts."
layoutable	boolean	Indicates whether the object supports the describeLayout () call (<code>true</code>) or not (<code>false</code>).
mergeable	boolean	Indicates whether the object can be merged with other objects of its type (<code>true</code>) or not (<code>false</code>). <code>true</code> for leads, contacts, and accounts.
mruEnabled	boolean	Indicates whether Most Recently Used (MRU) list functionality is enabled for the object (<code>true</code>) or not (<code>false</code>).
name	string	Name of the object. This is the same string that was passed in as the sObjectType parameter.
namedLayoutInfos	NamedLayoutInfo[]	The specific named layouts that are available for the objects other than the default layout.
networkScopeFieldName	string	The API name of the <code>networkScopeField</code> that scopes the entity to a community. For most entities, the value of this property is <code>null</code> .
queryable	boolean	Indicates whether the object can be queried via the query () call (<code>true</code>) or not (<code>false</code>).
recordTypeInfos	RecordTypeInfo[]	An array of the record types supported by this object. The user need not have access to all the returned record types to see them here.
replicable	boolean	Indicates whether the object can be replicated via the getUpdated () and getDeleted () calls (<code>true</code>) or not (<code>false</code>).
retrieveable	boolean	Indicates whether the object can be retrieved via the retrieve () call (<code>true</code>) or not (<code>false</code>).
searchable	boolean	Indicates whether the object can be searched via the search() call (<code>true</code>) or not (<code>false</code>).
searchLayoutable	boolean	Indicates whether search layout information can be retrieved via the describeSearchLayouts () call (<code>true</code>) or not (<code>false</code>).
triggerable	boolean	Indicates whether the object supports Apex triggers.
undeletable	boolean	Indicates whether an object can be undeleted using the undelete () call (<code>true</code>) or not (<code>false</code>).

Name	Type	Description
updateable	boolean	Indicates whether the object can be updated via the <code>update ()</code> call (<code>true</code>) or not (<code>false</code>).
urlDetail	string	URL to the read-only detail page for this object. Compare with <code>urlEdit</code> , which is read-write. Client applications can use this URL to redirect to, or access, the Salesforce user interface for standard and custom objects. To provide flexibility and allow for future enhancements, returned <code>urlDetail</code> values are dynamic. To ensure that client applications are forward compatible, it is recommended that they use this capability where possible. Note that, for objects for which a stable URL is not available, this field is returned empty.
urlEdit	string	URL to the edit page for this object. For example, the <code>urlEdit</code> field for the Account object returns <code>https://yourInstance.salesforce.com/{ID}/e</code> . Substituting the <code>{ID}</code> field for the current object ID will return the edit page for that specific account in the Salesforce user interface. Compare with <code>urlDetail</code> , which is read-only. Client applications can use this URL to redirect to, or access, the Salesforce user interface for standard and custom objects. To provide flexibility and allow for future enhancements, returned <code>urlDetail</code> values are dynamic. To ensure that client applications are forward compatible, it is recommended that they use this capability where possible. Note that, for objects for which a stable URL is not available, this field is returned empty.
urlNew	string	URL to the new/create page for this object. Client applications can use this URL to redirect to, or access, the Salesforce user interface for standard and custom objects. To provide flexibility and allow for future enhancements, returned <code>urlNew</code> values are dynamic. To ensure that client applications are forward compatible, it is recommended that they use this capability where possible. Note that, for objects for which a stable URL is not available, this field is returned empty.



Note: The properties with a Boolean value indicate whether certain API calls can be used for an object. However, other factors, such as permissions, also affect whether such operations can be performed on the object.

ActionOverride

ActionOverride provides details about an action that replaces the default action pages for an object. For example, an object could be configured to replace the new/create page with a custom page. This type is available in API version 32.0 and later.

Name	Type	Description
formFactor	string	Represents the environment to which the action override applies. For example, a <code>Large</code> value in this field represents the Lightning Experience desktop environment, and is valid only for Lightning Pages. This field is available in API version 37.0 and later.

Name	Type	Description
isAvailableInTouch	boolean	Indicates whether the action override is available in the Salesforce1 app (<code>true</code>) or not (<code>false</code>).
name	string	The name of the action that overrides the default action. For example, if the new/create page was overridden with a custom action, the name might be "New".
pageId	reference	The ID of the page for the action override.
url	string	The URL of the action override.

ChildRelationship

The name of the sObject that has a foreign key to the sObject being described.

Name	Type	Description
cascadeDelete	boolean	Indicates whether the child object is deleted when the parent object is deleted (<code>true</code>) or not (<code>false</code>).
childSObject	string	The name of the object on which there is a foreign key back to the parent sObject.
deprecatedAndHidden	boolean	Reserved for future use.
field	string	The name of the field that has a foreign key back to the parent sObject.
relationshipName	string	The name of the relationship, usually the plural of the value in childSObject .

Field

In the [DescribeSObjectResult](#), the `fields` property contains an array of `Field` objects. Each field represents a field in an API object. The array contains only the fields that the user can view, as defined by the user's field-level security settings.

Name	Type	Description
autonumber	boolean	Indicates whether this field is an autonumber field (<code>true</code>) or not (<code>false</code>). Analogous to a SQL <code>IDENTITY</code> type, autonumber fields are read only, non-createable text fields with a maximum length of 30 characters. Autonumber fields are read-only fields used to provide a unique ID that is independent of the internal object ID (such as a purchase order number or invoice number). Autonumber fields are configured entirely in the Salesforce user interface. The API provides access to this attribute so that client applications can determine whether a given field is an autonumber field.
byteLength	int	For variable-length fields (including binary fields), the maximum size of the field, in bytes.

Name	Type	Description
calculated	boolean	Indicates whether the field is a custom formula field (<code>true</code>) or not (<code>false</code>). Note that custom formula fields are always read-only.
caseSensitive	boolean	Indicates whether the field is case sensitive (<code>true</code>) or not (<code>false</code>).
controllerName	string	The name of the field that controls the values of this picklist. It only applies if <code>type</code> is <code>picklist</code> or <code>multipicklist</code> and <code>dependentPicklist</code> is true. See About Dependent Picklists . The mapping of controlling field to dependent field is stored in the <code>validFor</code> attribute of each <code>PicklistEntry</code> for this picklist. See validFor .
createable	boolean	Indicates whether the field can be created (<code>true</code>) or not (<code>false</code>). If <code>true</code> , then this field value can be set in a <code>create()</code> call.
custom	boolean	Indicates whether the field is a custom field (<code>true</code>) or not (<code>false</code>).
defaultValueCreate	boolean	Indicates whether this field is defaulted when created (<code>true</code>) or not (<code>false</code>). If <code>true</code> , then Salesforce implicitly assigns a value for this field when the object is created, even if a value for this field is not passed in on the <code>create()</code> call. For example, in the <code>Opportunity</code> object, the <code>Probability</code> field has this attribute because its value is derived from the <code>Stage</code> field. Similarly, the <code>Owner</code> has this attribute on most objects because its value is derived from the current user (if the <code>Owner</code> field is not specified).
defaultValueFormula	string	The default value specified for this field if the formula is not used. If no value has been specified, this field is not returned.
dependentPicklist	boolean	Indicates whether a picklist is a dependent picklist (<code>true</code>) where available values depend on the chosen values from a controlling field, or not (<code>false</code>). See About Dependent Picklists .
deprecatedAndHidden	boolean	Reserved for future use.
digits	int	For fields of type <code>integer</code> . Maximum number of digits. The API returns an error if an integer value exceeds the number of digits.
displayLocationInDecimal	boolean	Indicates how the value of a Geolocation custom field appears in the user interface. If <code>true</code> , the geolocation appears in decimal notation. If <code>false</code> , the geolocation appears as degrees, minutes, and seconds.
encrypted	boolean	<p> Note: This page is about Shield Platform Encryption, not Classic Encryption. What's the difference?</p> <p>Indicates whether this field is encrypted. This value only appears in the results of a <code>describeSObjects()</code> call when it is <code>true</code>; otherwise, it is omitted from the results. This field is available in API version 31.0 and later.</p>
extraTypeInfo	string	<p>If the field is a <code>textarea</code> field type, indicates if the text area is plain text (<code>plaintextarea</code>) or rich text (<code>richtextarea</code>).</p> <p>If the field is a <code>url</code> field type, if this value is <code>imageurl</code>, the URL references an image file. Available on standard fields on standard objects only, for example, <code>Account.photoUrl</code>, <code>Contact.photoUrl</code>, and so on.</p>

Name	Type	Description
		If the field is a <code>reference</code> field type, indicates the type of external object relationship. Available on external objects only. <ul style="list-style-type: none"> • <code>null</code>—lookup relationship • <code>externallookup</code>—external lookup relationship • <code>indirectlookup</code>—indirect lookup relationship
<code>filterable</code>	<code>boolean</code>	Indicates whether the field is filterable (<code>true</code>) or not (<code>false</code>). If <code>true</code> , then this field can be specified in the <code>WHERE</code> clause of a query string in a <code>query()</code> call.
<code>filteredLookupInfo</code>	<code>FilteredLookupInfo</code>	If the field is a <code>reference</code> field type with a lookup filter, <code>filteredLookupInfo</code> contains the lookup filter information for the field. If there is no lookup filter, or the filter is inactive, this field is <code>null</code> . This field is available in API version 31.0 and later.
<code>formula</code>	<code>string</code>	The formula specified for this field. If no formula is specified for this field, it is not returned.
<code>groupable</code>	<code>boolean</code>	Indicates whether the field can be included in the <code>GROUP BY</code> clause of a SOQL query (<code>true</code>) or not (<code>false</code>). See <code>GROUP BY</code> in the Salesforce SOQL and SOSL Reference Guide . Available in API version 18.0 and later.
<code>highScaleNumber</code>	<code>boolean</code>	Indicates whether the field stores numbers to 8 decimal places regardless of what's specified in the field details (<code>true</code>) or not (<code>false</code>). Used to handle currencies for products that cost fractions of a cent, in large quantities. If high-scale unit pricing isn't enabled in your organization, this field isn't returned. Available in API version 33.0 and later.
<code>htmlFormatted</code>	<code>boolean</code>	Indicates whether a field such as a hyperlink custom formula field has been formatted for HTML and should be encoded for display in HTML (<code>true</code>) or not (<code>false</code>). Also indicates whether a field is a custom formula field that has an <code>IMAGE</code> text function.
<code>idLookup</code>	<code>boolean</code>	Indicates whether the field can be used to specify a record in an <code>upsert()</code> call (<code>true</code>) or not (<code>false</code>).
<code>inlineHelpText</code>	<code>string</code>	The text that displays in the field-level help hover text for this field.
		 Note: This property is not returned unless at least one field on the object contains a value. When at least one field has field-level help, all fields on the object list the property with either the field-level help value or null for fields that have blank field-level help.
<code>label</code>	<code>string</code>	Text label that is displayed next to the field in the Salesforce user interface. This label can be localized.
<code>length</code>	<code>int</code>	For string fields, the maximum size of the field in Unicode characters (not bytes).
<code>mask</code>	<code>string</code>	Reserved for future use.

Name	Type	Description
maskType	string	Reserved for future use.
name	string	Field name used in API calls, such as <code>create()</code> , <code>delete()</code> , and <code>query()</code> .
nameField	boolean	<p>Indicates whether this field is a name field (<code>true</code>) or not (<code>false</code>). Used to identify the name field for standard objects (such as <code>AccountName</code> for an <code>Account</code> object) and custom objects. Limited to one per object, except where <code>FirstName</code> and <code>LastName</code> fields are used (such as in the <code>Contact</code> object).</p> <p>If a compound name is present, for example the <code>Name</code> field on a person account, <code>nameField</code> is set to <code>true</code> for that record. If no compound name is present, <code>FirstName</code> and <code>LastName</code> have this field set to <code>true</code>.</p>
namePointing	boolean	Indicates whether the field's value is the <code>Name</code> of the parent of this object (<code>true</code>) or not (<code>false</code>). Used for objects whose parents may be more than one type of object, for example a task may have an account or a contact as a parent.
nillable	boolean	Indicates whether the field is nillable (<code>true</code>) or not (<code>false</code>). A nillable field can have empty content. A non-nillable field must have a value in order for the object to be created or saved.
permissionable	boolean	Indicates whether <code>FieldPermissions</code> can be specified for the field (<code>true</code>) or not (<code>false</code>).
picklistValues	PicklistEntry[]	Provides the list of valid values for the picklist. Specified only if <code>restrictedPicklist</code> is <code>true</code> .
precision	int	For fields of type double. Maximum number of digits that can be stored, including all numbers to the left and to the right of the decimal point (but excluding the decimal point character).
relationshipName	string	The name of the relationship, if this is a master-detail relationship field.
relationshipOrder	int	<p>The type of relationship for a master-detail relationship field. Valid values are:</p> <ul style="list-style-type: none"> • 0 if the field is the primary relationship • 1 if the field is the secondary relationship
referenceTargetField	string	Applies only to indirect lookup relationships on external objects. Name of the custom field on the parent standard or custom object whose values are matched against the values of the child external object's indirect lookup relationship field. This matching is done to determine which records are related to each other. This field is available in API version 32.0 and later.
referenceTo	string[]	For fields that refer to other objects, this array indicates the object types of the referenced objects.
restrictedPicklist	boolean	Indicates whether the field is a restricted picklist (<code>true</code>) or not (<code>false</code>).
scale	int	For fields of type double. Number of digits to the right of the decimal point. The API silently truncates any extra digits to the right of the decimal point, but it returns a fault response if the number has too many digits to the left of the decimal point.

Name	Type	Description
soapType	SOAPType	See SOAPType for a list of allowable values.
sortable	boolean	Indicates whether a query can sort on this field (<code>true</code>) or not (<code>false</code>).
type	FieldType	See FieldType for a list of allowable values.
unique	boolean	Indicates whether the value must be unique (<code>true</code>) or not (<code>false</code>).
updateable	boolean	Indicates one of the following: <ul style="list-style-type: none"> Whether the field is updateable, (<code>true</code>) or not (<code>false</code>). If <code>true</code>, then this field value can be set in an update () call. If the field is in a master-detail relationship on a custom object, indicates whether the child records can be reparented to different parent records (<code>true</code>), <code>false</code> otherwise.
writeRequiresMasterRead	boolean	This field only applies to master-detail relationships. Indicates whether a user requires read sharing access (<code>true</code>) or write sharing access (<code>false</code>) to the parent record to insert, update, and delete a child record. In both cases, a user also needs Create, Edit, and Delete object permissions for the child object.

FieldType

In the `Field` object associated with the [DescribeSObjectResult](#), the `type` field can contain one of the following strings. For more information about field types, see [Field Types](#).

type Field Value	What the Field Object Contains
string	String values.
boolean	Boolean (<code>true</code> / <code>false</code>) values.
int	Integer values.
double	Double values.
date	Date values.
datetime	Date and time values.
base64	Base64-encoded arbitrary binary data (of type <code>base64Binary</code>). Used for Attachment , Document , and Scontrol objects.
ID	Primary key field for the object. For information on IDs, see ID Field Type .
reference	Cross-references to a different object. Analogous to a foreign key field in SQL.
currency	Currency values.
textarea	String that is displayed as a multiline text field.
percent	Percentage values.

type Field Value	What the Field Object Contains
phone	Phone numbers. Values can include alphabetic characters. Client applications are responsible for phone number formatting.
url	URL values. Client applications should commonly display these as hyperlinks. If <code>Field.extraTypeInfo</code> is <code>imageurl</code> , the URL references an image, and can be displayed as an image instead.
email	Email addresses.
combobox	Comboboxes, which provide a set of enumerated values and allow the user to specify a value not in the list.
picklist	Single-select picklists, which provide a set of enumerated values from which only one value can be selected.
multipicklist	Multi-select picklists, which provide a set of enumerated values from which multiple values can be selected.
anyType	Values can be any of these types: <code>string</code> , <code>picklist</code> , <code>boolean</code> , <code>int</code> , <code>double</code> , <code>percent</code> , <code>ID</code> , <code>date</code> , <code>dateTime</code> , <code>url</code> , or <code>email</code> .
location	Geolocation values, including latitude and longitude, for custom geolocation fields on custom objects.

FilteredLookupInfo

In the `Field` object associated with the [DescribeSObjectResult](#), the `filteredLookupInfo` field contains information about the lookup filter associated with the field.

This subtype is available in API version 31.0 and later.

Name	Type	Description
<code>controllingFields</code>	string[]	Array of the field's controlling fields when the lookup filter is dependent on the source object.
<code>dependent</code>	boolean	Indicates whether the lookup filter is dependent upon the source object (<code>true</code>) or not (<code>false</code>).
<code>optionalFilter</code>	boolean	Indicates whether the lookup filter is optional (<code>true</code>) or not (<code>false</code>).

SOAPType

The [DescribeSObjectResult](#) returns the `fields` property, which contains an array of fields whose value provides information about the object being described. One of those fields, `soapType`, contains one of the following string values. All of the values preceded by `xsd:` are XML schema primitive data types. For more information about the XML schema primitive data types, see the World Wide Web Consortium's publication *XML Schema Part 2: Data Types* at: <http://www.w3.org/TR/xmlschema-2/>.

Value	Description
tns:ID	Unique ID associated with an sObject . For information on IDs, see ID Field Type .
xsd:anyType	Can be ID, Boolean, double, integer, string, date, or dateTime.
xsd:base64Binary	Base 64-encoded binary data.
xsd:boolean	Boolean (<code>true</code> / <code>false</code>) values.
xsd:date	Date values.
xsd:dateTime	Date/time values.
xsd:double	Double values.
xsd:int	Integer values.
xsd:string	Character strings.

PicklistEntry

In the `Field` object associated with the [DescribeSObjectResult](#), the `picklistValues` field contains an array of `PicklistEntry` properties. Each `PicklistEntry` can contain any one of the following string values. For more information, see [Picklist Field Type](#).

Name	Type	Description
active	boolean	Indicates whether this item must be displayed (<code>true</code>) or not (<code>false</code>) in the drop-down list for the picklist field in the user interface.
validFor	byte[]	A set of bits where each bit indicates a controlling value for which this <code>PicklistEntry</code> is valid. See About Dependent Picklists .
defaultValue	boolean	Indicates whether this item is the default item (<code>true</code>) in the picklist or not (<code>false</code>). Only one item in a picklist can be designated as the default.
label	string	Display name of this item in the picklist.
value	string	Value of this item in the picklist.

About Dependent Picklists

A dependent picklist works in conjunction with a controlling field to filter its values. The value chosen in the controlling field affects the values available in the dependent picklist.

A dependent picklist can be any custom picklist or multi-select picklist field that displays available values based on the value selected in its corresponding controlling field. A controlling field can be any standard or custom picklist (with at least one and less than 200 values) or checkbox field whose values control the available values in one or more corresponding dependent fields.

In the following example, the controlling picklist `Beverage` has two values, which relate to the values of the dependent picklist `Beverage Variety`:

Beverage	Beverage Variety
Coffee	Decaffeinated

Beverage	Beverage Variety
	Regular
Tea	Chamomile
	Earl Grey
	English Breakfast

For each `PicklistEntry` that represents a value in a dependent picklist, the `validFor` attribute contains a set of bits. Each bit indicates a controlling field value for which the `PicklistEntry` is valid. Read the bits from left to right.

For more information on dependent picklists, see the “Dependent Picklists” topic in the Salesforce online help.

Sample Java Code for Dependent Picklists

```
public void dependentPicklistSample() {
    // inner class to decode a "validFor" bitset
    class Bitset {
        byte[] data;

        public Bitset(byte[] data) {
            this.data = data == null ? new byte[0] : data;
        }

        public boolean testBit(int n) {
            return (data[n >> 3] & (0x80 >> n % 8)) != 0;
        }

        public int size() {
            return data.length * 8;
        }
    }

    try {
        DescribeSObjectResult describeSObjectResult = connection.describeSObject("Case");
        Field[] fields = describeSObjectResult.getFields();
        // create a map of all fields for later lookup
        Map fieldMap = new HashMap();
        for (int i = 0; i < fields.length; i++) {
            fieldMap.put(fields[i].getName(), fields[i]);
        }
        for (int i = 0; i < fields.length; i++) {
            // check whether this is a dependent picklist
            if (fields[i].getDependentPicklist()) {
                // get the controller by name
                Field controller = (Field)fieldMap.get(fields[i].getControllerName());
                System.out.println("Field '" + fields[i].getLabel() + "' depends on '" +
                    controller.getLabel() + "'");

                PicklistEntry[] picklistValues = fields[i].getPicklistValues();
                for (int j = 0; j < picklistValues.length; j++) {
                    // for each PicklistEntry: list all controlling values for which it is valid
                    System.out.println("Item: '" + picklistValues[j].getLabel() +
                        " ValidFor: " + Integer.toBinaryString(picklistValues[j].validFor));
                }
            }
        }
    }
}
```

RecordTypeInfo

Base class for the old RecordTypeMapping object. This object contains all of the existing fields of RecordTypeMapping except layoutId and picklistForRecordType.

Name	Type	Description
available	boolean	Indicates whether this record type is available (<code>true</code>) or not (<code>false</code>). Availability is used to display a list of available record types to the user when they are creating a new record.
defaultRecordTypeMapping	boolean	Indicates whether this is the default record type mapping (<code>true</code>) or not (<code>false</code>).
master	boolean	Indicates whether this is the master record type (<code>true</code>) or not (<code>false</code>). The master record type is the default record type that's used when a record has no custom record type associated with it.
name	string	Name of this record type.
recordTypeId	ID	ID of this record type.

NamedLayoutInfo

The name of the named layout.

Name	Type	Description
name	string	Name of this layout.

describeSoftphoneLayout()

Retrieves layout information for a Salesforce CRM Call Center SoftPhone.

Syntax

```
DescribeSoftphoneLayoutResult[] = connection.describeSoftphoneLayout();
```

Usage

Use this call to obtain information about the layout of a SoftPhone. Use only in the context of Salesforce CRM Call Center; do not call directly from client programs.

Arguments

This call does not take any objects.

Response

The response is a DescribeSoftphoneLayoutResult object:

Name	Type	Description
callTypes	DescribeSoftphoneLayoutCallType[]	A set of attributes associated with each allowed call type. A call type may be Inbound, Outbound, or Internal.
id	ID	ID of layout. Note that layout objects are not exposed via the API.
name	string	Name of the call type: Inbound, Outbound, or Internal.

DescribeSoftphoneLayoutCallType

Each DescribeSoftphoneLayoutResult object contains one or more call types:

Name	Type	Description
infoFields	DescribeSoftphoneLayoutInfoField[]	A set of information field in the SoftPhone layout.
name	string	Name of the layout.

Name	Type	Description
screenPopOptions	DescribeSoftphoneScreenPopOption	Settings in the SoftPhone layout that specify how to display screen pops when the details of calls match or don't match existing records. This setting only displays for SoftPhone layouts associated with CTI 2.0 adapters or higher. See "Salesforce CTI Toolkit" in the Salesforce online help. This field is available in API version 18.0 and later.
screenPopsOpenWithin	string	Setting in the SoftPhone layout that specify whether to display screen pops in a new browser window or tab when the details of calls match or don't match existing records. This setting only displays for SoftPhone layouts associated with CTI 2.0 adapters or higher. See "Salesforce CTI Toolkit" in the Salesforce online help. This field is available in API version 18.0 and later.
sections	DescribeSoftphoneLayoutSection	A set of object names and the corresponding item name in the SoftPhone layout. There is one section for each object in a call type.

DescribeSoftphoneLayoutInfoField

An information field in the SoftPhone layout.

Name	Type	Description
name	string	The name of an information field in the SoftPhone layout that does not correspond to a Salesforce object. For example, caller ID may be specified in an information field. Information fields hold static information about the call type.

DescribeSoftphoneLayoutSection

Each call type returned in a `DescribeSoftphoneLayoutResult` object contains one section for each call type. Each section contains object-item pairs:

Name	Type	Description
entityApiName	string	The name of an object in the Salesforce application that corresponds to an item displayed in the SoftPhone layout, for example, a set of accounts or cases.
items	DescribeSoftphoneLayoutItem	A set of SoftPhone layout items.

DescribeSoftphoneLayoutItem

Each layout item corresponds to a record in Salesforce:

Name	Type	Description
itemApiName	string	The name of a record in the Salesforce application that corresponds to an item displayed in the SoftPhone layout, for example, the Acme account.

DescribeSoftphoneScreenPopOption

Each call type returned in a `DescribeSoftphoneLayoutResult` object contains one `screenPopOptions` field for each call type. Each `screenPopOptions` field contains details about screen pop settings:

Name	Type	Description
matchType	string	Setting on a SoftPhone layout to pop a screen for call details that match a single record, multiple records, or no records.
screenPopData	string	Setting on a SoftPhone layout for a specific object or page to pop for a call's <code>matchType</code> . For example, pop a specified Visualforce page when the details of a call match a record.
screenPopType	picklist	Setting that specifies how to pop a screen for a call's <code>matchType</code> . For example, pop a detail page or don't pop any page when the details of a call match a record.

Sample Code—Java

This sample describes the soft phone layout and writes its properties to the console. It then gets the allowed call types. For each call type, it gets its information fields, layout sections, and the layout items in the layout sections. It writes these values to the console.

```
public void describeSoftphoneLayout() {
    try {
        DescribeSoftphoneLayoutResult result =
            connection.describeSoftphoneLayout();
        System.out.println("ID of retrieved Softphone layout: " +
            result.getId());
        System.out.println("Name of retrieved Softphone layout: " +
            result.getName());
        System.out.println("\nContains following " +
            "Call Type Layouts\n");
        for (DescribeSoftphoneLayoutCallType type :
            result.getCallTypes()) {
            System.out.println("Layout for " + type.getName() +
                " calls");
            System.out.println("\tCall-related fields:");
            for (DescribeSoftphoneLayoutInfoField field :
                type.getInfoFields()) {
                System.out.println("\t\t" + field.getName());
            }
            System.out.println("\tDisplayed Objects:");
            for (DescribeSoftphoneLayoutSection section :
                type.getSections()) {
                System.out.println("\t\tFor entity " +
                    section.getEntityApiName() +
                    " following records are displayed:"
```

```
        );
        for (DescribeSoftphoneLayoutItem item :
            section.getItems()) {
            System.out.println("\t\t\t" + item.getItemApiName());
        }
    }
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample describes the soft phone layout and writes its properties to the console. It then gets the allowed call types. For each call type, it gets its information fields, layout sections, and the layout items in the layout sections. It writes these values to the console.

```
/// Demonstrates how to retrieve the layout information
/// for a Salesforce CRM Call Center SoftPhone
public void DescribeSoftphoneLayoutSample()
{
    try
    {
        DescribeSoftphoneLayoutResult dsplResult = binding.describeSoftphoneLayout();

        // Display the ID and Name of the layout
        Console.WriteLine("ID of retrieved Softphone layout: {0}", dsplResult.id);
        Console.WriteLine("Name of retrieved Softphone layout: {0}", dsplResult.name);

        // Display the contents of each Call Type
        Console.WriteLine("\nContains following Call Type Layouts\n");
        foreach (DescribeSoftphoneLayoutCallType dsplCallType in dsplResult.callTypes)
        {
            Console.WriteLine("Layout for {0} calls", dsplCallType.name);

            // Display the call-related fields contained in the call type
            Console.WriteLine("\tCall-related fields:");
            foreach (DescribeSoftphoneLayoutInfoField dsplInfoField
                in dsplCallType.infoFields)
            {
                Console.WriteLine("\t\t{0}", dsplInfoField.name);
            }

            // Display the objects that are included in the layout
            Console.WriteLine("\tDisplayed Objects:");
            foreach (DescribeSoftphoneLayoutSection dsplSection
                in dsplCallType.sections)
            {
                Console.WriteLine("\t\tFor entity {0} following records are displayed:",
                    dsplSection.entityApiName);
                foreach (DescribeSoftphoneLayoutItem dsplItem in dsplSection.items)
                {
                    Console.WriteLine("\t\t\t{0}", dsplItem.itemApiName);
                }
            }
        }
    }
}
```

```
        }
    }
}
catch (SoapException e)
{
    Console.WriteLine(e.Message);
    Console.WriteLine(e.StackTrace);
    Console.WriteLine(e.InnerException);
}
}
```

describeSoqlListViews ()

Retrieves the SOQL query and other information about a list view.

Syntax

```
connection.describeSoqlListViews (DescribeSoqlListViewsRequest request);
```

Usage

Use the `describeSoqlListViews ()` call to retrieve information about a list view, including the ID, the columns, and the SOQL query. This call is useful if you want to use the SOQL that drives an existing list view in your custom application. This call is available in API version 32.0 and later.

Sample Code—Java

```
public void example() throws Exception {
    DescribeSoqlListViewsRequest request =
createDescribeSoqlListViewsRequest(listViewId, null);
    this.getClient().describeSoqlListViews(request);
}
```

Arguments

Name	Type	Description
request	DescribeSoqlListViewsRequest	The fully qualified name or the ID of the list view and the object with which the list view is associated.

Response

A [DescribeSoqlListViewResult](#) object that contains one or more [DescribeSoqlListView](#) on page 2319 objects.

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

DescribeSoqlListView

Contains information about the specified list view, including the columns, sObject type, and SOQL query.

The `DescribeSoqlListView` object has the following properties:

Name	Type	Description
columns	ListViewColumn[]	The columns that are returned by the list view query.
id	ID	The list view's fully qualified ID.
orderBy	ListViewOrderBy[]	A list of fields to sort results by, the sort order, and the position to which null values should be sorted.
query	string	The fully composed SOQL query for the list view.
scope	string	A <code>filterScope</code> to use for limiting the results.
sObjectType	string	The object with which the list view is associated.
whereCondition	<code>SoqlWhereCondition</code>	Filter conditions on the list view. For example, for a list view called "Accounts this Week," <code>whereCondition</code> might contain "created data is within the last 7 days."

DescribeSoqlListViewParams

Use the `DescribeSoqlListViewParams` object with [describeSoqlListViews\(\)](#) to retrieve the SOQL from a list view.

The `DescribeSoqlListViewParams` object has the following properties:

Name	Type	Description
developerNameOrId	string	The list view's ID or fully qualified developer name.
sObjectType	string	The API name of the sObject for the list view.

DescribeSoqlListViewResult

Contains one or more `DescribeSoqlListView` objects, each of which contains information about one or more list views, including the ID, sObject type, columns, and SOQL query of each.

The `DescribeSoqlListViewResult` object has the following properties:

Name	Type	Description
<code>describeSoqlListViews</code>	DescribeSoqlListView[]	Information about one or more list views, including the ID, sObject type, columns, and SOQL query of each.

DescribeSoqlListViewsRequest

Use the `DescribeSoqlListViewsRequest` object with `describeSoqlListViews()` to retrieve information about a list view.

The `DescribeSoqlListViewsRequest` object has the following properties:

Name	Type	Description
<code>listViewParams</code>	DescribeSoqlListViewParams	A list of parameters that specify the list view to describe.

EDITIONS

Available in: Salesforce Classic

Available in:

ListViewColumn

Contains metadata about a single list view column.

The `ListViewColumn` object is returned by the `describeSoqlListViews()` and `executeListView()` calls. It has the following properties:

Name	Type	Description
<code>ascendingLabel</code>	string	The localized type-specific label for sorting the column in ascending order. For example: "A-Z" for a text field, or "Low to High" for a numeric field. Set to null if the column isn't sortable.
<code>descendingLabel</code>	string	The localized type-specific label for sorting the column in descending order. For example: "Z-A" for a text field, or "High to Low" for a numeric field. Set to null if the column is not sortable.
<code>fieldNameOrPath</code>	string	The field name or SOQL field path for the column.
<code>hidden</code>	boolean	If true, specifies that the column is not displayed, and is present only to support the display of other columns or other client-side logic.
<code>label</code>	string	The localized display label for the column.
<code>selectListItem</code>	string	The SOQL SELECT item for the column. The item might differ from the field name or path, due to display formatting (for example, <code>toLabel</code> for picklists).
<code>sortDirection</code>	<code>orderByDirection</code>	An enumerated value, one of the following if the column is sortable: <ul style="list-style-type: none"> • ascending • descending Set to null if the column is not sortable.
<code>sortIndex</code>	int	The zero-based index that indicates the column's position within a multilevel sort, or null if the records are not sorted by the column.
<code>sortable</code>	boolean	Whether the column is sortable, in which case it might be referenced in the <code>ExecuteListView</code> <code>orderBy</code> parameter.
<code>type</code>	<code>FieldType</code>	The column data type.

ListViewOrderBy

Use the `ListViewOrderBy` object with `executeListView()` to determine the order in which records are returned from a list view.

The `ListViewOrderBy` object is returned by the `describeSoqlListViews()` call, is an optional input to the `executeListView()` call, and has the following properties:

Name	Type	Description
<code>fieldNameOrPath</code>	string	The field name or SOQL path of the field on which to sort the records.
<code>nullsPosition</code>	<code>orderByNullsPosition</code>	An enumerated value that determines where nulls are sorted in the results: <ul style="list-style-type: none"> • first • last
<code>sortDirection</code>	<code>orderByDirection</code>	An enumerated value that determines the sort order of the results: <ul style="list-style-type: none"> • ascending • descending

describeTabs()

Returns information about the standard and custom apps available to the logged-in user, as listed in the Force.com app menu at the top of the page. An app is a set of tabs that works as a unit to provide application functionality. For example, two of the standard Salesforce apps are "Sales" and "Call Center."

Syntax

```
describeTabSetResult [] = connection.describeTabs();
```

Usage

Use the `describeTabs()` call to obtain information about the standard and custom apps to which the logged-in user has access. The `describeTabs()` call returns the minimum required metadata that can be used to render apps in another user interface. Typically this call is used by partner applications to render Salesforce data in another user interface.

For each app, the call returns the app name, the URL of the logo, whether or not it's the currently selected application for the user, and details about the tabs included in that app.

! **Important:** The `describeTabs()` call returns information only about tabs that display in the Salesforce user interface for the logged-in user. If a user clicks the All Tabs (+) tab and hides some tabs from his Salesforce user interface, those user-hidden tabs aren't included in the set of tabs returned by `describeTabs()`.

Use the `describeAllTabs()` call to obtain information about all the tabs that are available to the logged-in user.

For each tab, the call returns the tab name, the primary `sObject` that's displayed on the tab, whether it's a custom tab, and the URL for viewing that tab. Note that the "All Tabs" tab and Lightning Page tabs aren't included in the list of tabs.

Sample Code—Java

This sample calls `describeTabs()`, which returns an array of tab set results. Next, for each tab set result, which represents an app, it retrieves some of its properties and gets all the tabs for this app. It writes all retrieved properties to the console.

```
public void describeTabsSample() {
    try {
        // Describe tabs
        DescribeTabSetResult[] dtsrs = connection.describeTabs();
        System.out.println("There are " + dtsrs.length +
            " tab sets defined.");

        // For each tab set describe result, get some properties
        for (int i = 0; i < dtsrs.length; i++) {
            System.out.println("Tab Set " + (i + 1) + ":");

            DescribeTabSetResult dtsr = dtsrs[i];
            System.out.println("Label: " + dtsr.getLabel());
            System.out.println("\tLogo URL: " + dtsr.getLogoUrl());
            System.out.println("\tTab selected: " +
                dtsr.isSelected());

            // Describe the tabs for the tab set
            DescribeTab[] tabs = dtsr.getTabs();
            System.out.println("\tTabs defined: " + tabs.length);

            // Iterate through the returned tabs
            for (int j = 0; j < tabs.length; j++) {
                DescribeTab tab = tabs[j];
                System.out.println("\t\tTab " + (j + 1) + ":");

                System.out.println("\t\t\tName: " +
                    tab.getObjectName());
                System.out.println("\t\t\tLabel: " + tab.getLabel());
                System.out.println("\t\t\tURL: " + tab.getUrl());
                DescribeColor[] tabColors = tab.getColors();
                // Iterate through tab colors as needed
                DescribeIcon[] tabIcons = tab.getIcons();
                // Iterate through tab icons as needed
            }
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample calls `describeTabs()`, which returns an array of tab set results. Next, for each tab set result, which represents an app, it retrieves some of its properties and gets all the tabs for this app. It writes all retrieved properties to the console.

```
public void describeTabsSample() {
    try {
        // Describe tabs
        DescribeTabSetResult[] dtsrs = binding.describeTabs();
        Console.WriteLine("There are " + dtsrs.Length +
```

```
" tab sets defined.");

// For each tab set describe result, get some properties
for (int i = 0; i < dtsrs.Length; i++) {
    Console.WriteLine("Tab Set " + (i + 1) + ":");

    DescribeTabSetResult dtsr = dtsrs[i];
    Console.WriteLine("Label: " + dtsr.label);
    Console.WriteLine("\tLogo URL: " + dtsr.logoUrl);
    Console.WriteLine("\tTab selected: " +
        dtsr.selected);

    // Describe the tabs for the tab set
    DescribeTab[] tabs = dtsr.tabs;
    Console.WriteLine("\tTabs defined: " + tabs.Length);

    // Iterate through the returned tabs
    for (int j = 0; j < tabs.Length; j++) {
        DescribeTab tab = tabs[j];
        Console.WriteLine("\t\tTab " + (j + 1) + ":");

        Console.WriteLine("\t\tName: " +
            tab.sObjectName);
        Console.WriteLine("\t\tLabel: " + tab.label);
        Console.WriteLine("\t\tURL: " + tab.url);
        DescribeColor[] tabColors = tab.colors;
        // Iterate through tab colors as needed
        DescribeIcon[] tabIcons = tab.icons;
        // Iterate through tab icons as needed
    }
}
} catch (SoapException e) {
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
```

Arguments

None.

Response

[describeTabSetResult](#), [DescribeTab](#)

SEE ALSO:

[API Call Basics](#)

[Using the Partner WSDL](#)

[DescribeTab](#)

[describeTabSetResult](#)

describeTabSetResult

The [describeTabs \(\)](#) call returns an array of `DescribeTabSetResult` objects, which has the following properties:

Name	Type	Description
description	string	The description for this standard or custom app.
label	string	The display label for this standard or custom app. This value changes when tabs are renamed in the Salesforce user interface. See the Salesforce online help for more information.
logoUrl	string	A fully qualified URL to the logo image associated with the standard or custom app.
namespace	string	If this is a custom app, and a set of tabs in the custom app was installed as part of a managed package, the value of this attribute is the developer namespace prefix that the creator of the package chose when the Developer Edition organization was enabled to allow publishing a managed package. This attribute identifies elements of a Force.com AppExchange package.
selected	boolean	If <code>true</code> , then this standard or custom app is the user's currently selected app.
tabs	DescribeTab	An array of tabs that are displayed for the specified standard app or custom app.

DescribeColor

DescribeColor contains color metadata information for a tab. The [describeTabs \(\)](#) call returns an array of `DescribeTabSetResult` values. Each `DescribeTabSetResult` contains an array of `DescribeTab` values, and each `DescribeTab` contains an array of `DescribeColor` values.

Each `DescribeColor` is associated with a Salesforce user interface theme. For more information on themes, see [Identifying the Salesforce Style Your Users See](#) in the Visualforce Developer's Guide.

Color information can also be retrieved via the [describeTheme \(\)](#) and [describeGlobalTheme \(\)](#) calls. These calls return information on colors used for each object in your organization that can use theme icons and colors.

Name	Type	Description
color	string	The color described in web color RGB format—for example, "00FF00".
context	string	The color context, which determines whether the color is the main color (or primary) for the tab.
theme	string	The associated theme. Possible values include: <ul style="list-style-type: none"> • <i>theme2</i>—Theme used prior to Spring '10, called the "Salesforce Classic 2005 user interface theme" • <i>theme3</i>—Theme introduced in Spring '10, called the "Salesforce Classic 2010 user interface theme" • <i>theme4</i>—Theme introduced in Winter '14 for the mobile touchscreen version of Salesforce, and in Winter '16 for Lightning Experience • <i>custom</i>—Theme associated with a custom icon

Describelcon

Describelcon contains icon metadata information for a tab. The [describeTabs \(\)](#) call returns an array of [DescribeTabSetResult](#) values. Each [DescribeTabSetResult](#) contains an array of [DescribeTab](#) values, and each [DescribeTab](#) contains an array of [Describelcon](#) values.

Icon information can also be retrieved via the [describeTheme \(\)](#) and [describeGlobalTheme \(\)](#) calls. These calls return information on icons used for each object in your organization that can use theme icons and colors.

Name	Type	Description
contentType	string	The tab icon's content type, for example, "image/png."
height	int	The tab icon's height in pixels. If the icon content type is an SVG type, height and width values are not used.
theme	string	<p>The associated theme. Possible values include:</p> <ul style="list-style-type: none"> • <i>theme2</i>—Theme used prior to Spring '10, called the "Salesforce Classic 2005 user interface theme" • <i>theme3</i>—Theme introduced in Spring '10, called the "Salesforce Classic 2010 user interface theme" • <i>theme4</i>—Theme introduced in Winter '14 for the mobile touchscreen version of Salesforce, and in Winter '16 for Lightning Experience • <i>custom</i>—Theme associated with a custom icon
url	string	The fully qualified URL for this icon.
width	int	The tab icon's width in pixels. If the icon content type is an SVG type, height and width values are not used.

DescribeTab

The [describeTabs \(\)](#) call returns a [describeTabSetResult](#) object, of which [DescribeTab](#) is a property:

Name	Type	Description
colors	DescribeColor []	Array of color information used for a tab. This field is available in API version 29.0 and later.
custom	boolean	<code>true</code> if this is a custom tab, <code>false</code> if this is a standard tab.
iconUrl	string	The URL for the main 32x32 pixel icon for a tab. This icon appears next to the heading at the top of most pages. This icon URL corresponds to the 32x32 icon used for the Salesforce Classic 2010 user interface theme.
icons	Describelcon []	Array of icon information used for a tab. This field is available in API version 29.0 and later.
label	string	The display label for this tab.
miniIconUrl	string	The URL for the 16x16 pixel icon that represents a tab. This icon appears in related lists and other locations. This icon URL corresponds to the 16x16 icon used for the current Salesforce theme, introduced in Spring '10.

Name	Type	Description
name	string	The API name of the tab.
sobjectName	string	The name of the sObject that is primarily displayed on this tab (for tabs that display a particular SObject). For a list of objects, see Standard Objects .
url	string	A fully qualified URL for viewing this tab.

SEE ALSO:

[DescribeColor](#)[Describelcon](#)**describeTheme ()**

Returns information about themes available to the current logged-in user.

Syntax

```
DescribeThemeResult = connection.describeTheme(string sObjectType[]);
```

Usage

Use `describeTheme ()` to get current theme information for a given array of objects. Theme information consists of colors and icons for an object in Salesforce, used for a particular theme. For example, the `Merchandise__c` object might use the "computer32" icon and a primary tab color of red for the regular Salesforce application theme, and a different set of colors and icons for the mobile touchscreen version of Salesforce.

If you pass `null` instead of an array of objects, `describeTheme ()` returns theme information for all objects in your organization that use theme colors and icons.

Your client application must be logged in with sufficient access rights to retrieve theme information about your organization's data. For more information, see [Factors that Affect Data Access](#).

`describeTheme ()` is available in API version 29.0 and later.

Sample

This Java sample calls `describeTheme ()` to retrieve theme information for Account and Contact, and then iterates over the retrieved theme information.

```
public static void describeThemeExample() {
    try {
        // Get current themes
        DescribeTheme themeResult = connection.describeTheme(
            new String[] { "Account", "Contact" });
        DescribeThemeItem[] themeItems = themeResult.getThemeItems();
        for (int i = 0; i < themeItems.length; i++) {
            DescribeThemeItem themeItem = themeItems[i];
    }
}
```

```

        System.out.println("Theme information for object " + themeItem.getName());
        // Get color and icon info for each themeItem
        DescribeColor colors[] = themeItem.getColors();
        System.out.println("    Number of colors: " + colors.length);
        int k;
        for (k = 0; k < colors.length; k++) {
            DescribeColor color = colors[k];
            System.out.println("        For Color #" + k + ":");

            System.out.println("            Web RGB Color: " + color.getColor());
            System.out.println("            Context: " + color.getContext());
            System.out.println("            Theme: " + color.getTheme());
        }
        DescribeIcon icons[] = themeItem.getIcons();
        System.out.println("    Number of icons: " + icons.length);
        for (k = 0; k < icons.length; k++) {
            DescribeIcon icon = icons[k];
            System.out.println("        For Icon #" + k + ":");

            System.out.println("            ContentType: " + icon.getContentType());
            System.out.println("            Height: " + icon.getHeight());
            System.out.println("            Theme: " + icon.getTheme());
            System.out.println("            URL: " + icon.getUrl());
            System.out.println("            Width: " + icon.getWidth());
        }
    }
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}

```

Response

[DescribeThemeResult](#)

Faults

[UnexpectedErrorFault](#)

SEE ALSO:

[DescribeThemeResult](#)
[DescribeThemeltem](#)
[DescribeColor](#)
[Describelcon](#)

DescribeThemeResult

The `describeTheme()` and `describeGlobalTheme()` calls return `DescribeThemeResult`, which contains an array of `DescribeThemeltem` values.

Name	Type	Description
themes	DescribeThemeltem []	Array of themes. Theme information is provided for each object in the organization that can use theme icons and colors.

DescribeThemeltem

The [describeTheme \(\)](#) and [describeGlobalTheme \(\)](#) calls return [DescribeThemeResult](#), which contains an array of DescribeThemeltem values. Each DescribeThemeltem contains an array of colors and icons used for themes, and the name of the object the theme information applies to.

Name	Type	Description
colors	DescribeColor []	Array of colors.
icons	Describelcon []	Array of icons.
name	string	Name of the object that the theme colors and icons are associated with.

CHAPTER 11 Utility Calls

This topic describes API calls that your client applications can invoke to obtain the system timestamp, user information, and change user passwords.

 **Note:** For a list of general API calls, see [Core Calls](#). For a list of describe calls, see [Describe Calls](#).

The following table lists the API utility calls described in this topic:

Task / Call	Description
getServerTimestamp()	Retrieves the current system timestamp from the API.
getUserInfo()	Retrieves personal information for the user associated with the current session.
renderEmailTemplate()	Replaces merge fields in the text bodies of email templates with their corresponding Salesforce record IDs, even for polymorphic fields. The email template bodies and their corresponding <code>whoId</code> and <code>whatId</code> values are specified in the argument.
resetPassword()	Changes a user's password to a system-generated value.
sendEmail()	Immediately sends an email message.
sendEmailMessage()	Immediately sends up to 10 draft email messages.
setPassword()	Sets the specified user's password to the specified value.

Samples

The samples in this section are based on the enterprise WSDL file. They assume that you have already imported the WSDL file and created a connection. To learn how to do so, see the [Quick Start](#) tutorial.

getServerTimestamp()

Retrieves the current system timestamp (Coordinated Universal Time (UTC) time zone) from the API.

Syntax

```
GetServerTimestampResult timestamp = connection.getServerTimestamp();
```

Usage

Use `getServerTimestamp()` to obtain the current system timestamp from the API. You might do this if, for example, you need to use the exact timestamp for timing or data synchronization purposes. When you `create()` or `update()` an object, the API uses the system timestamp to update the `CreatedDate` and `LastModifiedDate` fields, respectively, in the object.

The `getServerTimestamp()` call always returns the timestamp in Coordinated Universal Time (UTC) time zone. However, your local system might automatically display the results in your local time based on your time zone settings.



Note: Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time zone. To determine how your development tool handles time values, refer to its documentation.

Sample Code—Java

This sample gets the server time and writes it to the console in the user's local time zone.

```
public void doGetServerTimestamp() {
    try {
        GetServerTimestampResult result = connection.getServerTimestamp();
        Calendar serverTime = result.getTimestamp();
        System.out.println("Server time is: "
            + serverTime.getTime().toString());
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample gets the server time and writes it to the console in the user's local time zone.

```
public void doGetServerTimestamp()
{
    try
    {
        GetServerTimestampResult result =
            binding.getServerTimestamp();
        DateTime serverTime = result.timestamp;
        Console.WriteLine("Server time is: " +
            serverTime.ToLocalTime().ToString());
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

Arguments

None.

Response

[getServerTimestampResult](#)

Fault

[UnexpectedErrorFault](#)

SEE ALSO:

[Utility Calls](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

getServerTimestampResult

The `getServerTimestamp()` call returns a `GetServerTimestampResult` object, which has the following properties:

Name	Type	Description
<code>timestamp</code>	<code>dateTime</code>	System timestamp of the API when the <code>getServerTimestamp()</code> call was executed.

getUserInfo()

Retrieves personal information for the user associated with the current session.

Syntax

```
getUserInfoResult result = connection.getUserInfo();
```

Usage

Use `getUserInfo()` to obtain personal information about the currently logged-in user. This convenience API call retrieves and aggregates common profile information that your client application can use for display purposes, performing currency calculations, and so on.

The `getUserInfo()` call applies only to the username under which your client application has logged in. To retrieve additional personal information not found in the `getUserInfoResult` object, you can call `retrieve()` on the `User` object and pass in the `userID` returned by this call. To retrieve personal information about other users, you could call `retrieve()` (if you know their user ID) or `query()` on the `User` object.

Sample Code—Java

This sample calls `getUserInfo()` and writes information about the current user to the console.

```
public void doGetUserInfo() {
    try {
        GetUserInfoResult result = connection.getUserInfo();
```

```
System.out.println("\nUser Information");
System.out.println("\tFull name: " + result.getUserFullName());
System.out.println("\tEmail: " + result.getUserEmail());
System.out.println("\tLocale: " + result.getUserLocale());
System.out.println("\tTimezone: " + result.getUserTimeZone());
System.out.println("\tCurrency symbol: " + result.getCurrencySymbol());
System.out.println("\tOrganization is multi-currency: " +
    result.isOrganizationMultiCurrency());
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
}
```

Sample Code—C#

This sample calls `getUserInfo()` and writes information about the current user to the console.

```
public void doGetUserInfo()
{
    try
    {
        GetUserInfoResult result = binding.getUserInfo();
        Console.WriteLine("\nUser Information");
        Console.WriteLine("\tFull name: " + result.userFullName);
        Console.WriteLine("\tEmail: " + result.userEmail);
        Console.WriteLine("\tLocale: " + result.userLocale);
        Console.WriteLine("\tTimezone: " + result.userTimeZone);
        Console.WriteLine("\tCurrency symbol: " + result.currencySymbol);
        Console.WriteLine("\tOrganization is multi-currency: " +
            result.organizationMultiCurrency);
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

Arguments

None.

Response

`getUserInfoResult`

Fault

[UnexpectedErrorFault](#)

SEE ALSO:

[Utility Calls](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

getUserInfoResult

The [getUserInfo\(\)](#) call returns a `GetUserInfoResult` object.

Name	Type	Description
accessibilityMode	boolean	Available in API version 7.0 and later. Indicates whether user interface modifications for the visually impaired are on (<code>true</code>) or off (<code>false</code>). The modifications facilitate the use of screen readers such as JAWS.
currencySymbol	string	Currency symbol to use for displaying currency values. Applicable only when <code>organizationMultiCurrency</code> is <code>false</code> .
organizationId	ID	ID of the organization. Allows third-party tools to uniquely identify individual organizations in Salesforce, which is useful for retrieving billing or organization-wide setup information.
organizationMultiCurrency	boolean	Indicates whether the user's organization uses multiple currencies (<code>true</code>) or not (<code>false</code>).
organizationName	string	Name of the user's organization or company.
orgDefaultCurrencyIsoCode	string	Default currency ISO code. Applicable only when <code>organizationMultiCurrency</code> is <code>false</code> . When the logged-in user creates any objects that have a currency ISO code, the API uses this currency ISO code if it is not explicitly specified in the create() call.
profileID	ID	ID of the profile associated with the role currently assigned to the user.
roleID	ID	Role ID of the role currently assigned to the user.
sessionSecondsValid	int	The number of seconds before the session expires, starting from the last update time. Available in API version 21.0 and later.
userDefaultCurrencyIsoCode	string	Default currency ISO code. Applicable only when <code>organizationMultiCurrency</code> is <code>true</code> . When the logged-in user creates any objects that have a currency ISO code, the API uses this currency ISO code if it is not explicitly specified in the create() call.
userEmail	string	User's email address.
userFullName	string	User's full name.
userID	ID	User ID.

Name	Type	Description
userLanguage	string	User's language, which controls the language for labels displayed in an application. String is 2-5 characters long. The first two characters are always an ISO language code, for example "fr" or "en." If the value is further qualified by country, then the string also has an underscore (_) and another ISO country code, for example "US" or "UK. For example, the string for the United States is "en_US", and the string for French Canadian is "fr_CA." For a list of the languages that Salesforce supports, see the Salesforce online help topic "What languages does Salesforce support?"
userLocale	string	User's locale, which controls the formatting of dates and choice of symbols for currency. The first two characters are always an ISO language code, for example "fr" or "en." If the value is further qualified by country, then the string also has an underscore (_) and another ISO country code, for example "US" or "UK. For example, the string for the United States is "en_US", and the string for French Canadian is "fr_CA."
userName	string	User's login name.
userTimeZone	string	User's time zone.
userType	string	Type of user license assigned to the Profile associated with the user.
userUiSkin	string	Available in API version 7.0 and later. Possible values are: <ul style="list-style-type: none">• theme3—if the user is using the Salesforce Classic 2010 user interface theme, also known as the <i>Aloha</i> interface• theme2—if the user is using the Salesforce Classic 2005 user interface theme• theme1—if the user is using the oldest user interface theme (obsolete) In the online app, this look and feel setting is configurable from Setup by entering <i>User Interface</i> in the Quick Find box, then selecting User Interface . See User Interface Themes .

renderEmailTemplate()

Replaces merge fields in the text bodies of email templates with their corresponding Salesforce record IDs, even for polymorphic fields. The email template bodies and their corresponding `whoId` and `whatId` values are specified in the argument.

Syntax

```
RenderEmailTemplateResult = connection.renderEmailTemplate(RenderEmailTemplateRequest []
renderRequests);
```

Usage

The `renderEmailTemplate()` call is equivalent to rendering merge fields when sending an email with a custom template through the `sendEmail()` call.

The `renderEmailTemplate()` call can take up to 10 `RenderEmailTemplateRequest` elements in its array argument, and each `RenderEmailTemplateRequest` can contain up to 10 template bodies. Each request is independent from the other requests in the array—an error in one request doesn't affect the other requests. Similarly, an error in one template body doesn't cause an error in other text bodies within the same request.

The `renderEmailTemplate()` call substitutes a merge field with the value of either the `whatId` or `whoId` in `RenderEmailTemplateRequest`:

- If the merge field references a non-human object, it's replaced with the corresponding value of `whatId`. For example, if a merge field references an account or opportunity, the `whatId` value is substituted.
- If the merge field references a human object, it's replaced with the corresponding value of `whoId`. For example, if a merge field references a contact, lead, or user, the `whoId` value is substituted.

The `whatId` and `whoId` field values of `RenderEmailTemplateRequest` are validated for each request. If the `whatId` doesn't reference a valid what ID (a non-human object), or the `whoId` doesn't reference a valid who ID (a human object), an error is set for the request.

If no value is found for a merge field, the `renderEmailTemplate()` call returns an error for the corresponding template body. For example, if a merge field is replaced with a blank or `null`, an error is returned.

Sample Code—Java

In this sample, the `renderEmailTemplate()` call substitutes all contact merge fields with the value from the specified `whoId` argument. Similarly, the call substitutes the opportunity merge field (`{ !Opportunity.Name }`) with the specified `whatId` value. The second template body in this sample has an incorrect merge field (`{ !Contact.SNARF }`), which causes an error on the second template. However, the entire template rendering request is successful.

```
public void renderTemplates(String whoId, String whatId)
    throws ConnectionException, RemoteException, MalformedURLException {
    // Array of three template bodies.
    // The second template body generates an error.
    final String[] TEMPLATE_BODIES = new String[] {
        "This is a good template body {!Contact.Name}",
        "This is a bad template body {!Opportunity.Name} {!Contact.SNARF} ",
        "This is another good template body {!Contact.Name}";

    // Create request and add template bodies, whatId, and whoId.
    RenderEmailTemplateRequest req = new RenderEmailTemplateRequest();
    req.setTemplateBodies(TEMPLATE_BODIES);
    req.setWhatId(whatId);
    req.setWhoId(whoId);
    // An array of results is returned, one for each request.
    // We only have one request.
    RenderEmailTemplateResult[] results = connection.renderEmailTemplate(
        new RenderEmailTemplateRequest[] { req });
    if (results != null) {
        // Check results for our one and only request.
        // Check request was processed successfully, and if not, print the errors.
        if (!results[0].isSuccess()) {
            System.out.println(
```

```
"The following errors were encountered while rendering email templates:");
for (Error err : results[0].getErrors()) {
    System.out.println(err.getMessage());
}
} else {
    // Check results for each body template and print merged body
    RenderEmailTemplateBodyResult[] bodyResults = results[0].getBodyResults();
    for( Integer i=0;i<bodyResults.length;i++) {
        RenderEmailTemplateBodyResult result = bodyResults[i];
        if (result.isSuccess()) {
            System.out.println("\nMerged body: \n" + result.getMergedBody());
        } else {
            System.out.println("\nErrors were found for body[" + i + "]: ");
            for (RenderEmailTemplateError err : result.getErrors()) {
                System.out.println(err.getMessage() + " - Field name: "
                    + err.getFieldName());
            }
        }
    }
}
```

Let's say you run this sample by specifying a valid contact ID for the first argument (`whoId`) and `null` for the second argument (`whatId`). The second template has two errors set. The first error is for the incorrect merge field, and the second error is because the opportunity merge field (`{!Opportunity.Name}`) wasn't found due to the null `whatId`. The response looks like the following.

Merged body:
This is a good template body Howard Jones

Errors were found for body[1]:
Field Contact.SNARF does not exist. Check spelling. - Field name: Contact.SNARF
No value for the field: Opportunity.Name - Field name: Opportunity.Name

Merged body:
This is another good template body Howard Jones

RenderEmailTemplateRequest

Name	Type	Description
templateBodies	string[]	An array of text bodies that can contain merge fields, such as { !Account.Phone } or { !Contact.Name }.
whatId	reference	References a non-human object, such as an account, an opportunity, a campaign, a case, or a custom object. The whatId is polymorphic, which means that it's an ID that can refer to more than one type of object, such as a case or an opportunity.
whoId	reference	References a human object, such as a lead or a contact. The whoId is polymorphic, which means that it is equivalent to a contact's ID or a lead's ID.

Fault

The `renderEmailTemplate()` can return any of these API status codes.

`EMAIL_TEMPLATE_FORMULA_ERROR`
`EMAIL_TEMPLATE_MERGEFIELD_ACCESS_ERROR`
`EMAIL_TEMPLATE_MERGEFIELD_ERROR`
`EMAIL_TEMPLATE_MERGEFIELD_VALUE_ERROR`
`EMAIL_TEMPLATE_PROCESSING_ERROR`

RenderEmailTemplateResult

Contains status and error information for a request processed by the `renderEmailTemplate()` call, including individual results of rendered email templates.

Name	Type	Description
<code>bodyResults</code>	<code>RenderEmailTemplateBodyResult[]</code>	Contains status and error information for each template body that <code>renderEmailTemplate()</code> processed in a request, including merged body text of templates.
<code>errors</code>	<code>Error[]</code>	Contains one or more errors that occurred when <code>renderEmailTemplate()</code> rendered a request.
<code>success</code>	<code>boolean</code>	Indicates whether a request was successfully processed (<code>true</code>) or not (<code>false</code>).

RenderEmailTemplateBodyResult

Contains status and error information for each template body that `renderEmailTemplate()` processed in a request, including merged body text of templates.

Name	Type	Description
<code>errors</code>	<code>RenderEmailTemplateError[]</code>	Contains one or more errors that are associated with a template body that <code>renderEmailTemplate()</code> processed.
<code>mergedBody</code>	<code>string</code>	The text of the template body with the merge fields replaced with their corresponding values from Salesforce objects. The <code>whatId</code> and <code>whoId</code> fields on the request reference the Salesforce objects to use. The <code>mergedBody</code> field is populated only when the rendering of the template was successful (<code>success</code> is equal to <code>true</code>). If <code>success</code> is equal to <code>false</code> , <code>mergedBody</code> is null.
<code>success</code>	<code>boolean</code>	Indicates whether a template body was successfully rendered (<code>true</code>) or not (<code>false</code>).

RenderEmailTemplateError

An error that occurred when [renderEmailTemplate\(\)](#) processed a template body.

Name	Type	Description
fieldName	string[]	The merge field that affected the error condition.
message	string	Error message text.
offset	int	The offset in the template body text of the merge field that caused the error. The offset is calculated as the number of characters from the start of the body text. The offset is -1 if it can't be determined because of insufficient contextual information.
statusCode	StatusCode	A code that characterizes the error. The full list of status codes is available in the WSDL file for your organization (see Generating the WSDL File for Your Organization).

resetPassword()

Changes a user's password to a temporary, system-generated value.

Syntax

```
string password = connection.resetPassword(ID userID);
```

Usage

Use [resetPassword\(\)](#) to request that the API change the password of a [User](#) or [SelfServiceUser](#), and return a system-generated password string of random letters and numbers. Use [setPassword\(\)](#) instead if you want to set the password to a specific value.

Your client application must be logged in with sufficient access rights to change the password for the specified user. For more information, see [Factors that Affect Data Access](#).

For information on IDs, see [ID Field Type](#).

Sample Code—Java

This sample resets the password for the user specified by the `userID` parameter. It calls `resetPassword()` with this ID and gets the temporary password from the call result. It writes this temporary password to the console and returns it.

```
public String doResetPassword(String userId) {
    String result = "";
    try {
        ResetPasswordResult rpr = connection.resetPassword(userId);
        result = rpr.getPassword();
        System.out.println("The temporary password for user ID " + userId
            + " is " + result);
    } catch (ConnectionException ce) {
```

```

        ce.printStackTrace();
    }
    return result;
}

```

Sample Code—C#

This sample resets the password for the user specified by the `userId` parameter. It calls `resetPassword()` with this ID and gets the temporary password from the call result. It writes this temporary password to the console and returns it.

```

public String doResetPassword(String userId)
{
    String result = "";
    try
    {
        ResetPasswordResult rpr = binding.resetPassword(userId);
        result = rpr.password;
        Console.WriteLine("The temporary password for user ID " + userId + " is " +
            result);
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
    return result;
}

```

Arguments

Name	Type	Description
userId	ID	ID of the User or SelfServiceUser whose password you want to reset. For information on IDs, see ID Field Type .

Response

Name	Type	Description
password	string	New password generated by the API. Once the user logs in with this password, they will be asked to provide a new password. This password is temporary, meaning that it cannot be reused once the user has set his or her new password.

Fault

[InvalidIdFault](#)

UnexpectedErrorFault

SEE ALSO:

[Utility Calls](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

sendEmail()

Immediately sends an email message.

Syntax

For single email messages:

```
SendEmailResult = connection.sendEmail(SingleEmailMessage emails[]);
```

For mass email messages:

```
SendEmailResult = connection.sendEmail(MassEmailMessage emails[]);
```

Usage

Use this call with Force.com AppExchange applications, custom applications, or other applications outside of Salesforce to send individual and mass email. The email can include all standard email attributes (such as subject line and blind carbon copy address), use Salesforce email templates, and be in plain text or HTML format. You can use Salesforce to track the status of HTML email, including the date the email was sent, first opened, last opened, and the total number of times it was opened. (See “Tracking HTML Email” in the Salesforce online help for more information.)

The email address of the logged-in user is inserted in the `From Address` field of the email header. All return email and out-of-office replies go to the logged-in user. If bounce management is enabled and `SingleEmailMessage.targetObjectId` or `MassEmailMessage.targetObjectIds` is set, bounces are processed by Salesforce automatically, and the appropriate records are updated; otherwise, they go to the logged-in user. Bounce management works for contacts and leads only.



Note:

- Single email messages sent with this call count against the sending organization's daily single email limit. When this limit is reached, `sendEmail()` calls using `SingleEmailMessage` are rejected, and the user receives a `SINGLE_EMAIL_LIMIT_EXCEEDED` error code. However, single emails sent through the application are allowed.
- Mass email messages sent with this call count against the sending organization's daily mass email limit. When this limit is reached, `sendEmail()` calls using `MassEmailMessage` are rejected, and the user receives a `MASS_MAIL_LIMIT_EXCEEDED` error code.
- Starting in API version 35.0, you can enforce or ignore the **Email Opt Out** setting for contacts or leads with the `optOutPolicy` field of `SingleEmailMessage`. The `optOutPolicy` field applies to recipients in the To, CC, and BCC lists of the email. By default and in earlier versions, `SingleEmailMessage` ignores the **Email Opt Out** setting of recipients and the email is sent to all recipients. When using `MassEmailMessage`, the **Email Opt Out** setting of the recipients is always enforced—emails aren't sent to recipients that have opted out and are sent to all other recipients.

`SingleEmailMessage` has an optional field called `OrgWideEmailAddressId`. This is an object ID to an `OrgWideEmailAddress` object. If `OrgWideEmailAddressId` is set, the `OrgWideEmailAddress.displayName` field is used in the email header, instead of the

logged-in user's `Display Name`. The sending email address in the header is also set to the field defined in `OrgWideEmailAddress.Address`.

-  **Note:** If both the `DisplayName` in an `OrgWideEmailAddress` and `senderDisplayName` are defined, the user receives a [DUPLICATE_SENDER_DISPLAY_NAME](#) error.

Sample Code—Java

This sample creates an email message and sets its fields, including the To, CC and BCC recipients, subject, and body text. It also sets a recipient to the ID of the logged-in user using the `setTargetObjectId` method, which causes the email to be sent to the email address of the specified user. The sample creates an attachment and sends the email message with the attachment. Finally, it writes a status message or an error message, if any, to the console.

```
public void doSendEmail() {
    try {
        EmailFileAttachment efa = new EmailFileAttachment();
        byte[] fileBody = new byte[1000000];
        efa.setBody(fileBody);
        efa.setFileName("attachment");
        SingleEmailMessage message = new SingleEmailMessage();
        message.setBccAddresses(new String[] {
            "someone@salesforce.com"
        });
        message.setCcAddresses(new String[] {
            "person1@salesforce.com", "person2@salesforce.com", "003xx00000a1b2cAAC"
        });
        message.setBccSender(true);
        message.setEmailPriority>EmailPriority.High;
        message.setReplyTo("person1@salesforce.com");
        message.setSaveAsActivity(false);
        message.setSubject("This is how you use the " + "sendEmail method.");
        // We can also just use an id for an implicit to address
        GetUserInfoResult guir = connection.getUserInfo();
        message.setTargetObjectId(gui.getUserId());
        message.setUseSignature(true);
        message.setPlainTextBody("This is the humongous body "
            + "of the message.");
        EmailFileAttachment[] efas = { efa };
        message.setFileAttachments(efas);
        message.setToAddresses(new String[] { "person3@salesforce.com" });
        SingleEmailMessage[] messages = { message };
        SendEmailResult[] results = connection.sendEmail(messages);
        if (results[0].isSuccess()) {
            System.out.println("The email was sent successfully.");
        } else {
            System.out.println("The email failed to send: "
                + results[0].getErrors()[0].getMessage());
        }
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

This example shows how to send an email with the opt-out setting enforced. Recipients are specified by their IDs. The `SendEmailOptOutPolicy.FILTER` option causes the email to be sent only to recipients that haven't opted out from email.

```
SingleEmailMessage message = new SingleEmailMessage();
// Set recipients to two contact IDs.
// Replace IDs with valid record IDs in your org.
message.setToAddresses(new String[] { "003D000000QDexS", "003D000000QDFW5" });
message.setOptOutPolicy(SendEmailOptOutPolicy.FILTER);
message.setSubject("Opt Out Test Message");
message.setPlainTextBody("This is the message body.");
SingleEmailMessage[] messages = { message };
SendEmailResult[] results = connection.sendEmail(messages);
if (results[0].isSuccess()) {
    System.out.println("The email was sent successfully.");
} else {
    System.out.println("The email failed to send: "
        + results[0].getErrors()[0].getMessage());
}
```

Sample Code—C#

This sample creates an email message and sets its fields, including the To, CC and BCC recipients, subject, and body text. It also sets a recipient to the ID of the logged-in user using the `setTargetObjectId` method, which causes the email to be sent to the email address of the specified user. The sample creates an attachment and sends the email message with the attachment. Finally, it writes a status message or an error message, if any, to the console.

```
public void doSendEmail()
{
    try
    {
        EmailFileAttachment efa = new EmailFileAttachment();
        byte[] fileBody = new byte[1000000];
        efa.body = fileBody;
        efa.fileName = "attachment";
        SingleEmailMessage message = new SingleEmailMessage();
        message.setBccAddresses(new String[] {
            "someone@salesforce.com"
        });
        message.setCcAddresses(new String[] {
            "person1@salesforce.com", "person2@salesforce.com", "003xx00000a1b2cAAC"
        });
        message.bccSender = true;
        message.emailPriority = EmailPriority.High;
        message.replyTo = "person1@salesforce.com";
        message.saveAsActivity = false;
        message.subject = "This is how you use the " + "sendEmail method.";
        // We can also just use an id for an implicit to address
        GetUserInfoResult quir = binding.getUserInfo();
        message.targetObjectId = quir.userId;
        message.useSignature = true;
        message.plainTextBody = "This is the humongous body "
            + "of the message.";
        EmailFileAttachment[] efas = { efa };
        message.fileAttachments = efas;
    }
}
```

```

message.toAddresses = new String[] { "person3@salesforce.com" };
SingleEmailMessage[] messages = { message };
SendEmailResult[] results = binding.sendEmail(messages);
if (results[0].success)
{
    Console.WriteLine("The email was sent successfully.");
}
else
{
    Console.WriteLine("The email failed to send: "
        + results[0].errors[0].message);
}
}
catch (SoapException e)
{
    Console.WriteLine("An unexpected error has occurred: " +
        e.Message + "\n" + e.StackTrace);
}
}
}

```

This example shows how to send an email with the opt-out setting enforced. Recipients are specified by their IDs. The `SendEmailOptOutPolicy.FILTER` option causes the email to be sent only to recipients that haven't opted out from email.

```

SingleEmailMessage message = new SingleEmailMessage();
// Set recipients to two contact IDs.
// Replace IDs with valid record IDs in your org.
message.toAddresses = new String[] { "003D000000QdexS", "003D000000QdfW5" };
message.optOutPolicy = SendEmailOptOutPolicy.FILTER;
message.subject = "Opt Out Test Message";
message.plainTextBody = "This is the message body.";
SingleEmailMessage[] messages = { message };
SendEmailResult[] results = binding.sendEmail(messages);
if (results[0].success)
{
    Console.WriteLine("The email was sent successfully.");
} else {
    Console.WriteLine("The email failed to send: "
        + results[0].errors[0].message);
}
}

```

BaseEmail

The following table contains the arguments used in both single and mass email.

 **Note:** If templates are not being used, all email content must be in plain text, HTML, or both.

Name	Type	Description
bccSender	boolean	Indicates whether the email sender receives a copy of the email that is sent. For a mass mail, the sender is only copied on the first email sent.  Note: If the BCC compliance option is set at the organization level, the user cannot add BCC addresses on standard messages. The following error

Name	Type	Description
		code is returned: <code>BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED</code> . Contact your Salesforce representative for information on BCC compliance.
saveAsActivity	boolean	Optional. The default value is <code>true</code> , meaning the email is saved as an activity. This argument only applies if the recipient list is based on <code>targetObjectId</code> or <code>targetObjectIds</code> . If HTML email tracking is enabled for the organization, you will be able to track open rates.
useSignature	boolean	Indicates whether the email includes an email signature if the user has one configured. The default is <code>true</code> , meaning if the user has a signature it is included in the email unless you specify <code>false</code> .
emailPriority	picklist	<p>Optional. The priority of the email.</p> <ul style="list-style-type: none"> • Highest • High • Normal • Low • Lowest <p>The default is Normal.</p>
replyTo	string	Optional. The email address that receives the message when a recipient replies. This cannot be set if you are using a Visualforce email template that specifies a <code>replyTo</code> value.
subject	string	Optional. The email subject line. If you are using an email template and attempt to override the subject line, an error message is returned.
templateId	ID	The ID of the template to be merged to create this email.
senderDisplayName	string	Optional. The name that appears on the <code>From</code> line of the email. This cannot be set if the object associated with a <code>OrgWideEmailAddressId</code> for a <code>SingleEmailMessage</code> has defined its <code>DisplayName</code> field.

SingleEmailMessage

The following table contains the arguments single email uses in addition to the base email arguments.

Name	Type	Description
bccAddresses	string[]	<p>Optional. An array of blind carbon copy (BCC) addresses or object IDs of the contacts, leads, and users you're sending the email to. The maximum number of <code>bccAddresses</code> is 25. This argument is allowed only when a template is not used.</p> <p>You can specify opt-out email options with the <code>optOutPolicy</code> field only for those recipients who were added by their IDs.</p>

Name	Type	Description
		<p>Email addresses are verified to ensure that they have the correct format and haven't been marked as bounced.</p> <p>If the BCC COMPLIANCE option is set at the organization level, the user cannot add BCC addresses on standard messages. The following error code is returned:</p> <p><code>BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED</code>.</p> <p>All emails must have a recipient value in at least one of the following fields:</p> <ul style="list-style-type: none"> • <code>toAddresses</code> • <code>ccAddresses</code> • <code>bccAddresses</code> • <code>targetObjectId</code>
<code>ccAddresses</code>	<code>string[]</code>	<p>Optional. An array of carbon copy (CC) addresses or object IDs of the contacts, leads, and users you're sending the email to. The maximum number of <code>ccAddresses</code> is 25. This argument is allowed only when a template is not used.</p> <p>You can specify opt-out email options with the <code>optOutPolicy</code> field only for those recipients who were added by their IDs.</p> <p>Email addresses are verified to ensure that they have the correct format and haven't been marked as bounced.</p> <p>All emails must have a recipient value in at least one of the following fields:</p> <ul style="list-style-type: none"> • <code>toAddresses</code> • <code>ccAddresses</code> • <code>bccAddresses</code> • <code>targetObjectId</code>
<code>charset</code>	<code>string</code>	<p>Optional. The character set for the email. If this value is null, the user's default value is used. Unavailable if specifying <code>templateId</code> because the template specifies the character set.</p>
<code>documentAttachments</code>	<code>ID[]</code>	<p>Deprecated. Use <code>entityAttachments</code> instead. Optional. An array listing the ID of each <code>Document</code> you want to attach to the email.</p>
<code>fileAttachments</code>	<code>EmailFileAttachment[]</code>	<p>Optional. An array listing the file names of the binary and text files you want to attach to the email. You can attach multiple files as long as the total size of all attachments does not exceed 10 MB.</p>
<code>entityAttachments</code>	<code>ID[]</code>	<p>Optional. Array of IDs of <code>Document</code> or <code>ContentVersion</code> items to attach to the email.</p> <p>This field is available in API version 35.0 and later.</p>

Name	Type	Description
htmlBody	string	Optional. The HTML version of the email, specified by the sender. The value is encoded according to the specification associated with the organization.
inReplyTo	string	Optional. The In-Reply-To field of the outgoing email. Identifies the emails to which this one is a reply (parent emails). Contains the parent emails' Message-IDs. See RFC2822 – Internet Message Format .
optOutPolicy	SendEmailOptOutPolicy (enumeration of type string)	Optional. If you added recipients by ID instead of email address and the Email Opt Out option set, this field determines the behavior of the <code>sendEmail()</code> call. The opt-out settings for recipients added by their email addresses aren't checked and those recipients always receive the email. Possible values of the <code>SendEmailOptOutPolicy</code> enumeration are: <ul style="list-style-type: none"> SEND (default)—The email is sent to all recipients and the Email Opt Out option of the recipients is ignored. FILTER—No email is sent to the recipients that have the Email Opt Out option set and emails are sent to the other recipients. REJECT—if any of the recipients have the Email Opt Out option set, <code>sendEmail()</code> throws an error and no email is sent. <p>This field is available in API version 35.0 and later.</p>
orgWideEmailAddressId	ID	Optional. The object ID of the OrgWideEmailAddress associated with the outgoing email. <code>OrgWideEmailAddress.DisplayName</code> cannot be set if the <code>senderDisplayName</code> field is already set.
plainTextBody	string	Optional. The text version of the email, specified by the sender.
references	string	Optional. The References field of the outgoing email. Identifies an email thread. Contains the parent emails' Message-ID and References fields and possibly In-Reply-To fields. See RFC2822 – Internet Message Format .
targetObjectId	ID	Optional. The object ID of the contact, lead, or user the email will be sent to. The object ID you enter sets the context and ensures that merge fields in the template contain the correct data All emails must have a recipient value in at least one of the following fields: <ul style="list-style-type: none"> toAddresses ccAddresses bccAddresses targetObjectId

Name	Type	Description
toAddresses	string[]	<p>Optional. An array of email addresses or object IDs of the contacts, leads, or users you're sending the email to. The maximum number of <code>toAddresses</code> is 100. This argument is allowed only when a template is not used.</p> <p>You can specify opt-out email options with the <code>optOutPolicy</code> field only for those recipients who were added by their IDs.</p> <p>Email addresses are verified to ensure that they have the correct format and haven't been marked as bounced.</p> <p>All emails must have a recipient value in at least one of the following fields:</p> <ul style="list-style-type: none"> • <code>toAddresses</code> • <code>ccAddresses</code> • <code>bccAddresses</code> • <code>targetObjectId</code>
treatBodiesAsTemplate	boolean	<p>Optional. If set to <code>true</code>, the subject, plain text, and HTML text bodies of the email are treated as template data. The merge fields are resolved using the <code>renderEmailTemplate()</code> call. Default is <code>false</code>.</p> <p>This field is available in API version 35.0 and later.</p>
treatTargetObjectAsRecipient	boolean	<p>Optional. If set to <code>true</code>, the <code>targetObjectId</code> (a contact, lead, or user) is the recipient of the email. If set to <code>false</code>, the <code>targetObjectId</code> is supplied as the <code>whoId</code> field for template rendering but isn't a recipient of the email. The default is <code>true</code>.</p> <p>This field is available in API version 35.0 and later. In prior versions, the <code>targetObjectId</code> is always a recipient of the email.</p>
whatId	ID	<p>Optional. If you specify a contact for the <code>targetObjectId</code> field, you can specify a <code>whatId</code> as well. This field helps to further ensure that merge fields in the template contain the correct data. The value must be one of the following types:</p> <ul style="list-style-type: none"> • Account • Asset • Campaign • Case • Contract • Opportunity • Order • Product • Solution • Custom

MassEmailMessage

The following table contains the arguments mass email uses in addition to the base email arguments.

Name	Type	Description
description	string	A value used internally to identify the object in the mass email queue.
targetObjectIds	ID[]	An array of object IDs of the contacts, leads, or users the email will be sent to. The object IDs you enter set the context and ensure that merge fields in the template contain the correct data. The objects must be of the same type (either all contacts, all leads, or all users). You can list up to 250 IDs per email. If you specify a value for the <code>targetObjectIds</code> field, optionally specify a <code>whatId</code> as well to set the email context to a user, contact, or lead. This ensures that merge fields in the template contain the correct data.
whatIds	ID[]	<p>Optional. If you specify an array of contacts for the <code>targetObjectIds</code> field, you can specify an array of <code>whatIds</code> as well. This helps to further ensure that merge fields in the template contain the correct data. The values must be one of the following types:</p> <ul style="list-style-type: none"> • Contract • Case • Opportunity • Product <p>If you specify <code>whatIds</code>, specify one for each <code>targetObjectId</code>; otherwise, you will receive an <code>INVALID_ID_FIELD</code> error.</p>

EmailFileAttachment

The following table contains properties that the EmailFileAttachment uses in the SingleEmailMessage object to specify attachments passed in as part of the request, as opposed to a [Document](#) passed in using the `documentAttachments` argument.

Property	Type	Description
body	base64	The attachment itself.
contentType	string	Optional. The attachment's Content-Type.
fileName	string	The name of the file to attach.
inline	boolean	Optional. Specifies a Content-Disposition of inline (<code>true</code>) or attachment (<code>false</code>). In most cases, inline content is displayed to the user when the message is opened. Attachment content requires user action to be displayed.

Response

[SendEmailResult](#)

Fault

The following API status codes can be returned. Also, `sendEmail()` can return other errors when rendering email templates. See [renderEmailTemplate\(\) Faults](#).

```
BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED
BCC_SELF_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED
DUPLICATE_SENDER_DISPLAY_NAME
EMAIL_ADDRESS_BOUNCED
EMAIL_NOT_PROCESSED_DUE_TO_PRIOR_ERROR
EMAIL_OPTED_OUT
ERROR_IN_MAILER
INSUFFICIENT_ACCESS_ON_CROSS_REFERENCE_ENTITY
INVALID_CONTENT_TYPE
INVALID_EMAIL_ADDRESS
INVALID_ID_FIELD
INVALID_MESSAGE_ID_REFERENCE
INVALID_SAVE_AS_ACTIVITY_FLAG
LIMIT_EXCEEDED
MALFORMED_ID
MASS_MAIL_LIMIT_EXCEEDED
NO_MASS_MAIL_PERMISSION
REQUIRED_FIELD_MISSING
SINGLE_EMAIL_LIMIT_EXCEEDED
TEMPLATE_NOT_ACTIVE
UNVERIFIED_SENDER_ADDRESS
```

SendEmailResult

The `sendEmail()` call returns a list of `SendEmailResult` objects. Each `SendEmailResult` object has the following properties:

Name	Type	Description
success	boolean	<p>If sending single email: Indicates whether the email was successfully accepted for delivery by the message transfer agent (<code>true</code>) or not (<code>false</code>). Even if <code>success = true</code>, it does not mean the intended recipients received the email, as it could have bounced or been blocked by a spam blocker. Also, even if the email is successfully accepted for delivery by the message transfer agent, there can still be errors in the error array related to individual addresses within the email.</p> <p>If sending mass email: Indicates whether the email was successfully added to the queue for processing (<code>true</code>) or not (<code>false</code>). Even if the email was added to the</p>

Name	Type	Description
		queue, there can still be processing errors that prevent delivery to the intended recipients.
SendEmailError[]	Error[]	If an error occurred during the <code>sendEmail()</code> call, a list of <code>SendEmailError</code> objects is returned. For single email, errors indicate that Salesforce wasn't able to deliver the email. For mass email, errors indicate that the email wasn't added to the queue for processing.

SendEmailError

SendEmailError can have the following attributes:

Name	Type	Description
Fields	Field[]	Reserved for future use. Array of one or more field names. Identifies which fields in the object, if any, affected the error condition.
Message	string	Error message text.
StatusCode	statusCode	A code that characterizes the error. The full list of status codes is available in the WSDL file for your organization .
TargetObjectId	ID	The object ID of the target for which the error occurred.

 **Note:** If an error occurs that prevents `sendEmail()` from sending the email to one or more targets, each `TargetObjectId` for those targets has an associated error in `SendEmailResult`. A `TargetObjectId` that does not have an associated error in `SendEmailResult` indicates the email was sent to the target. If `SendEmailResult` has an error that does not have an associated `TargetObjectId`, no email was sent.

The following is an example of how to parse through a resulting set for errors:

```

Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
email.setToAddresses(new String[] { 'admin@acme.com' });
email.setSubject('my subject');
email.setPlainTextBody('plain text body');
List<Messaging.SendEmailResult> results =
    Messaging.sendEmail(new Messaging.Email[] { email });
if (!results.get(0).isSuccess()) {
    System.StatusCode statusCode = results.get(0).getErrors()[0].getStatusCode();
    String errorMessage = results.get(0).getErrors()[0].getMessage();
}

```

sendEmailMessage()

Immediately sends up to 10 draft email messages.

Syntax

For Enterprise SOAP:

```
SendEmailResult[] = connection.sendEmailMessage( String[] draftEmailIds);
```

For Partner SOAP:

```
SendEmailResult[] = connection.sendEmailMessage( ID[] draftEmailIds);
```

Usage

Use this call with Force.com AppExchange applications, custom applications, or other applications outside of Salesforce to send up to 10 draft email messages. The messages can include all standard email attributes (such as subject line and blind carbon copy address), use Salesforce email templates, and be in plain text or HTML format. You can use Salesforce to track the status of HTML email, including the date the email was sent, first opened, last opened, and the total number of times it was opened. (See "Tracking HTML Email" in the Salesforce online help for more information.)

The email address of the logged-in user is inserted in the `From Address` field of the email header. All return email and out-of-office replies go to the logged-in user. If bounce management is enabled and `SingleEmailMessage.targetObjectId` or `MassEmailMessage.targetObjectIds` is set, bounces are processed by Salesforce automatically, and the appropriate records are updated; otherwise, they go to the logged-in user. Bounce management works for contacts and leads only.



Note:

- Email messages sent with this call count against the sending organization's daily single email limit. When this limit is reached, `sendEmailMessage()` calls using `SingleEmailMessage` are rejected, and the user receives a `SINGLE_EMAIL_LIMIT_EXCEEDED` error code. However, single emails sent through the application are allowed.
- Mass email messages sent with this call count against the sending organization's daily mass email limit. When this limit is reached, `sendEmail()` calls using `MassEmailMessage` are rejected, and the user receives a `MASS_MAIL_LIMIT_EXCEEDED` error code.
- The `AllOrNone` header is not honored by this call. `sendEmailMessage()` returns partial success even if the `AllOrNone` header is set to `true`.

Sample Code—Java

This sample creates a case and a draft email message, and sets the message fields, including the `From`, `To`, `CC`, and `BCC` recipients, subject, and body text. It also creates an attachment and sends the email message with the attachment. Finally, it writes a status message or an error message, if any, to the console.

```
public void doSendEmail() {
    try {
        //Create a case
        Case theCase = new Case();
        theCase.setSubject("Sample Case");
        SaveResult[] saveResult = connection.create(new SObject[] { theCase });
        String caseId = saveResult[0].getId();

        //Create a draft EmailMessage
        EmailMessage message = new EmailMessage();
        message.setParentId(theCase.getId());
        message.setBccAddress("bcc@email.com");
```

```
message.setCcAddress("cc1@salesforce.com; cc2@email.com");
message.setSubject("This is how you use the sendEmailMessage method.");
message.setFromAddress("from@email.com");
message.setFromName("Sample Code");
message.setTextBody("This is the text body of the message.");
message.setStatus("5"); // "5" means Draft
message.setToAddress("to@email.com");
saveResult = connection.create(new SObject[] { message });
String emailMessageId = saveResult[0].getId();

//Create an attachment for the draft EmailMessage
Attachment att = new Attachment();
byte[] fileBody = new byte[1000000];
att.setBody(fileBody);
att.setName("attachment");
att.setParentId(emailMessageId);
connection.create(new SObject[] { att });

//Send the draft EmailMessage
SendEmailResult[] results = connection.sendEmailMessage(messages);
if (results[0].isSuccess()) {
    System.out.println("The email was sent successfully.");
} else {
    System.out.println("The email failed to send: " +
        results[0].getErrors()[0].getMessage());
}
} catch (ConnectionException ce) {
    ce.printStackTrace();
}
```

Arguments

None.

Response

SendEmailResult[]

Fault

BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED
BCC_SELF_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED
EMAIL_NOT_PROCESSED_DUE_TO_PRIOR_ERROR
ERROR_IN_MAILER
INSUFFICIENT_ACCESS_ON_CROSS_REFERENCE_ENTITY
INVALID_CONTENT_TYPE
INVALID_EMAIL_ADDRESS
INVALID_ID_FIELD

INVALID_MESSAGE_ID_REFERENCE
LIMIT_EXCEEDED
MALFORMED_ID
REQUIRED_FIELD_MISSING
SINGLE_EMAIL_LIMIT_EXCEEDED
TEMPLATE_NOT_ACTIVE
UNVERIFIED_SENDER_ADDRESS

setPassword()

Sets the specified user's password to the specified value.

Syntax

```
SetPasswordResult setPasswordResult = connection.setPassword(ID userID, string password);
```

Usage

Use [setPassword\(\)](#) to change the password of a [User](#) or [SelfServiceUser](#) to a value that you specify. For example, a client application might prompt a user to specify a different password, and then invoke [setPassword\(\)](#) to change the user's password. Use [resetPassword\(\)](#) instead if you want to reset the password with a random value generated by the API.

Your client application must be logged in with sufficient access rights to change the password for the specified user. For more information, see [Factors that Affect Data Access](#).

For information on IDs, see [ID Field Type](#).

This call can use the session ID returned in [LoginResult](#) if the password has expired. For more information, see [passwordExpired](#).

Sample Code—Java

This sample accepts user ID and password parameters, which it uses in the `setPassword()` call to set the password of the specified user.

```
public void doSetPassword(String userId, String newPasswd) {
    try {
        SetPasswordResult result = connection.setPassword(userId, newPasswd);
        System.out.println("The password for user ID " + userId + " changed to "
            + newPasswd);
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Sample Code—C#

This sample accepts user ID and password parameters, which it uses in the `setPassword()` call to set the password of the specified user.

```
public void doSetPassword(String userId, String newPassword)
{
    try
    {
        SetPasswordResult result = binding.setPassword(userId, newPassword);
        Console.WriteLine("The password for user ID " + userId + " changed to "
            + newPassword);
    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}
```

Arguments

Name	Type	Description
userID	ID	ID of the User or SelfServiceUser whose password you want to reset. For information on IDs, see ID Field Type .
password	string	New password to use for the specified user.

Response

`SetPasswordResult` (empty)

Fault

`InvalidIdFault`

`UnexpectedErrorFault`

SEE ALSO:

[resetPassword\(\)](#)

[Utility Calls](#)

https://wiki.developerforce.com/index.php/Sample_SOAP_Messages

CHAPTER 12 SOAP Headers

The API provides SOAP headers to client applications.

Header	Description
AllOrNoneHeader	Specifies whether a call rolls back all changes unless all records are processed successfully. This header is available in API version 20.0 and later.
AllowFieldTruncationHeader	Specifies the truncation behavior for some field types in API version 15.0 and later.
AssignmentRuleHeader	Specifies the assignment rule to use when creating or updating an Account , Case , or Lead .
CallOptions	Specifies the call options for an API request.
DisableFeedTrackingHeader	Specifies whether the changes made in the current call are tracked in feeds.
EmailHeader	Sends an email notification when a request is processed. Provides equivalent functionality for the Salesforce user interface.
LimitInfoHeader	A response header returned from calls to the SOAP API. This header returns limit information for the organization. Use this header to monitor your API limits as you make calls against the organization.
LocaleOptions	Specifies the language of the labels returned. The value must be a valid user locale (language and country), such as <code>de_DE</code> or <code>en_GB</code> . For more information on locales, see the Language field on the CategoryNodeLocalization object.
LoginScopeHeader	Specifies the organization ID so that you can authenticate Self-Service users for your organization using the login() call.
MruHeader	Indicates whether to update the list of most recently used items (<code>true</code>) or not (<code>false</code>).
OwnerChangeOptions	Specifies ownership of attachments and notes.
PackageVersionHeader	Specifies the package version for each installed managed package in API version 16.0 and later.
QueryOptions	Specifies the batch size for query results.
SessionHeader	Specifies the session ID returned from the login server after a successful login() .
UserTerritoryDeleteHeader	Specifies a user to whom open opportunities are assigned when the current owner is removed from a territory.

AllOrNoneHeader

Allows a call to roll back all changes unless all records are processed successfully.

Without the AllOrNoneHeader header, records without errors are committed, while records with errors are marked as failed in the call results. This header is available in API version 20.0 and later.

Even if the header is enabled, it's still necessary to inspect the `success` field in the call result for each record to identify records with errors. Each `success` field contains `true` or `false` indicating whether the call was processed successfully.

If there is an error associated with at least one record, the `errors` field in the call result for the record gives more information on the error. If other records in the same call have no errors, their `errors` fields indicate that they were rolled back due to other errors.

API Calls

`create()`, `delete()`, `undelete()`, `update()`, `upsert()`

Fields

Element Name	Type	Description
allOrNone	boolean	If <code>true</code> , any failed records in a call cause all changes for the call to be rolled back. Record changes aren't committed unless all records are processed successfully. The default is <code>false</code> . Some records can be processed successfully while others are marked as failed in the call results.

Sample Code—Java

This sample shows how to use the `AllOrNoneHeader`. It attempts to create two contacts. The second contact doesn't have all required fields set and causes a failure on creation. Next, the sample sets the `allOrNone` field to `true`, and then attempts to create the contacts. Creating one of the contacts results in an error, so the entire transaction is rolled back and no contacts are created.

```
public void allOrNoneHeaderSample() {
    try {
        // Create the first contact.
        SObject[] sObjects = new SObject[2];
        Contact contact1 = new Contact();
        contact1.setFirstName("Robin");
        contact1.setLastName("Van Persie");

        // Create the second contact. This contact doesn't
        // have a value for the required
        // LastName field so the create will fail.
        Contact contact2 = new Contact();
        contact2.setFirstName("Ashley");
        sObjects[0] = contact1;
        sObjects[1] = contact2;

        // Set the SOAP header to roll back the create unless
    }
}
```

```
// all contacts are successfully created.  
connection.setAllOrNoneHeader(true);  
// Attempt to create the two contacts.  
SaveResult[] sr = connection.create(sObjects);  
for (int i = 0; i < sr.length; i++) {  
    if (sr[i].isSuccess()) {  
        System.out.println("Successfully created contact with id: " +  
            sr[i].getId() + ".");  
    }  
    else {  
        // Note the error messages as the operation was rolled back  
        // due to the all or none header.  
        System.out.println("Error creating contact: " +  
            sr[i].getErrors()[0].getMessage());  
        System.out.println("Error status code: " +  
            sr[i].getErrors()[0].getStatusCode());  
    }  
}  
}  
} catch (ConnectionException ce) {  
    ce.printStackTrace();  
}  
}
```

AllowFieldTruncationHeader

Specifies that for some fields, when a string is too large, the operation fails. Without the header, strings for these fields are truncated.

The `AllowFieldTruncationHeader` header affects the following datatypes:

- anyType, if it represents one of the other datatypes in this list
 - email
 - encryptedstring
 - multipicklist
 - phone
 - picklist
 - string
 - textarea

In API versions previous to 15.0, if a value for one of the listed fields is too large, the value is truncated.

For API version 15.0 and later, if a value is too large, the operation fails and the fault code `STRING_TOO_LONG` is returned.

`AllowFieldTruncationHeader` allows you to specify that the previous behavior, truncation, be used instead of the new behavior in API versions 15.0 and later.

This header has no effect in versions 14.0 and earlier.

API Calls

`convertLead()`, `create()`, `merge()`, `process()`, `undelete()`, `update()`, and `upsert()`

Apex: `executeanonymous()`

Fields

Element Name	Type	Description
allowFieldTruncation	boolean	<p>If <code>true</code>, truncate field values that are too long, which is the behavior in API versions 14.0 and earlier.</p> <p>Default is <code>false</code>: no change in behavior. If a <code>string</code> or <code>textarea</code> value is too large, the operation fails and the fault code <code>STRING_TOO_LONG</code> is returned.</p> <p>The following list shows the field types affected by truncation and this header:</p> <ul style="list-style-type: none"> • <code>anyType</code>, if it represents one of the other datatypes in this list • <code>email</code> • <code>encryptedstring</code> • <code>multipicklist</code> • <code>phone</code> • <code>picklist</code> • <code>string</code> • <code>textarea</code>

Sample Code—Java

To create an account with a name that is too long for the `Name` field, use the `AllowFieldTruncation` header.

This sample:

1. Creates an `Account` object with a name that exceeds the field limit of 255 characters.
2. Sends the create call, which fails because of the name field length.
3. Sets the `AllowFieldTruncationHeader` to `true` and retries the account creation, which succeeds.

```
public void allowFieldTruncationSample() {
    try {
        Account account = new Account();
        // Construct a string that is 256 characters long.
        // Account.Name's limit is 255 characters.
        String accName = "";
        for (int i = 0; i < 256; i++) {
            accName += "a";
        }
        account.setName(accName);
        // Construct an array of SObjects to hold the accounts.
        SObject[] sObjects = new SObject[1];
        sObjects[0] = account;
        // Attempt to create the account. It will fail in API version 15.0
        // and above because the account name is too long.
        SaveResult[] results = connection.create(sObjects);
        System.out.println("The call failed because: "
            + results[0].getErrors()[0].getMessage());
        // Now set the SOAP header to allow field truncation.
        connection.setAllowFieldTruncationHeader(true);
    }
}
```

```
// Attempt to create the account now.  
results = connection.create(sObjects);  
System.out.println("The call: " + results[0].isSuccess());  
} catch (ConnectionException ce) {  
    ce.printStackTrace();  
}  
}
```

AssignmentRuleHeader

The `AssignmentRuleHeader` must be specified in the `create()` or `update()` call of a `Case` or `Lead` for the specified assignment rule to be applied, and it must be specified in the `update()` call of an `Account` for the territory assignment rules to be applied.

API Calls

`create()`, `merge()`, `update()`, `upsert()`

Fields

Element Name	Type	Description
assignmentRuleId	ID	<p>The ID of a specific assignment rule to run for the Case or Lead. The assignment rule can be active or inactive. The ID can be retrieved by querying the AssignmentRule object. If specified, do not specify <code>useDefaultRule</code>. This element is ignored for accounts, because all territory assignment rules are applied.</p> <p>If the value is not in correct ID format (15-character or 18-character Salesforce ID), the call fails and a MALFORMED_ID exception is returned.</p>
useDefaultRule	boolean	<p>If <code>true</code> for a Case or Lead, uses the default (active) assignment rule for a Case or Lead. If specified, do not specify an <code>assignmentRuleId</code>. If <code>true</code> for an Account, all territory assignment rules are applied, and if <code>false</code>, no territory assignment rules are applied.</p>

Sample Code

For a code example, see [Lead](#).

SEE ALSO:

AssignmentRule

CallOptions

Specifies the options needed to work with a specific client. This header is only available for use with the [Partner WSDL](#).

API Calls

The `defaultNamespace` element supports the following calls: `create()`, `merge()`, `queryAll()`, `query()`, `queryMore()`, `retrieve()`, `search()`, `update()`, and `upsert()`.

The `client` element supports all of the above calls, plus the following: `convertLead()`, `login()`, `delete()`, `describeGlobal()`, `describeLayout()`, `describeTabs()`, `describeSObject()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `process()`, `undelete()`, `getServerTimestamp()`, `getUserInfo()`, `setPassword()`, and `resetPassword()`.

Fields

Element Name	Type	Description
<code>client</code>	<code>string</code>	A string that identifies a client.
<code>defaultNamespace</code>	<code>string</code>	<p>A string that identifies a developer namespace prefix. Use this field to resolve field names in managed packages without having to fully specify the <code>fieldName</code> everywhere. For example, if the developer namespace prefix is <code>battle</code>, and you have a custom field in your package called <code>botId</code>, you can set this header, and then queries such as the following will succeed:</p> <pre>query("SELECT id, botId__c from Account");</pre> <p>In this case the actual field queried is the <code>battle__botId__c</code> field. Using this field allows you to write client code without having to specify the namespace prefix. Without this field specified, the full name of the field would have to be used for the query to succeed. In the example above, you would have to specify <code>battle__botId__c</code>. Note that if this field is set, and the query specifies the namespace as well, the response will not include the prefix. For example, if you set this header to <code>battle</code>, and issue a query like <code>query("SELECT id, battle__botId__c from Account")</code>, the response would use a <code>botId__c</code> element, not a <code>battle__botId__c</code> element. Describe calls ignore this header, so there will be no ambiguity between fields with namespace prefixes and customer fields of the same name without the prefix.</p>

Sample Code—C#

This sample shows how to use the `CallOptions` header. It sets a client ID and a developer namespace prefix, which is used to resolve field names in managed packages. Next, the sample logs the specified user in.

```
public void CallOptionsSample()
{
```

```

// Web Reference to the imported Partner WSDL.
APISamples.partner.SforceService partnerBinding;

string username = "USERNAME";
string password = "PASSWORD";

// The real Client ID will be an API Token provided by salesforce.com
// to partner applications following a security review.
// For more details, see the Security Review FAQ in the online help.
string clientId = "SampleCaseSensitiveToken/100";

partnerBinding = new SforceService();
partnerBinding.CallOptionsValue = new CallOptions();
partnerBinding.CallOptionsValue.client = clientId;

// Optionally, if a developer namespace prefix has been registered for
// your Developer Edition organization, it may also be specified.
string prefix = "battle";
partnerBinding.CallOptionsValue.defaultNamespace = prefix;

try
{
    APISamples.partner.LoginResult lr =
        partnerBinding.login(username, password);
}
catch (SoapException e)
{
    Console.WriteLine(e.Code);
    Console.WriteLine(e.Message);
}
}
}

```

DisableFeedTrackingHeader

Specifies that changes made in the current call are tracked in feeds.

Use this header if you want to process many records without tracking the changes in various feeds related to the records. This header is available if the Chatter feature is enabled for your organization.

API Calls

`convertLead(), create(), delete(), merge(), process(), undelete(), update(), upsert()`

Fields

Element Name	Type	Description
disableFeedTracking	boolean	If <code>true</code> , the changes made in the current call are not tracked in feeds. The default is <code>false</code> .

Sample Code—Java

This sample shows how to use the `DisableFeedTrackingHeader`. It sets this header to `true` to disable feed tracking and then creates many account records in bulk.

```
public void disableFeedTrackingHeaderSample() {  
    try {  
        // Insert a large number of accounts.  
        SObject[] sObjects = new SObject[500];  
        for (int i = 0; i < 500; i++) {  
            Account a = new Account();  
            a.setName("my-account-" + i);  
            sObjects[i] = a;  
        }  
        // Set the SOAP header to disable feed tracking to avoid generating a  
        // large number of feed items because of this bulk operation.  
        connection.setDisableFeedTrackingHeader(true);  
        // Perform the bulk create. This won't result in 500 feed items, which  
        // would otherwise be generated without the DisableFeedTrackingHeader.  
        SaveResult[] sr = connection.create(sObjects);  
        for (int i = 0; i < sr.length; i++) {  
            if (sr[i].isSuccess()) {  
                System.out.println("Successfully created account with id: " +  
                    sr[i].getId() + ".");  
            } else {  
                System.out.println("Error creating account: " +  
                    sr[i].getErrors()[0].getMessage());  
            }  
        }  
    } catch (ConnectionException ce) {  
        ce.printStackTrace();  
    }  
}
```

SEE ALSO:

- [NewsFeed](#)
- [EntitySubscription](#)

DuplicateRuleHeader

Determines options for using duplicate rules to detect duplicate records. Duplicate rules are part of the Duplicate Management feature.

API Calls

`create()`, `update()`, `upsert()`

Fields

Element Name	Type	Description
allowSave	boolean	Set to <code>true</code> to save the duplicate record. Set to <code>false</code> to prevent the duplicate record from being saved.
includeRecordDetails	boolean	Set to <code>true</code> to get fields and values for records detected as duplicates. Set to <code>false</code> to get only record IDs for records detected as duplicates.
runAsCurrentUser	boolean	Set to <code>true</code> to make sure that sharing rules for the current user are enforced when duplicate rules run. Set to <code>false</code> to use the sharing rules specified in the class for the request. If no sharing rules are specified, Apex code runs in system context and sharing rules for the current user are not enforced.

Java Sample

This sample shows how to use the `DuplicateRuleHeader` to set options for using duplicate rules. To see the entire sample application, see [DuplicateResult](#).

```
_DuplicateRuleHeader header = new _DuplicateRuleHeader();
    header.setAllowSave(false);
    header.setIncludeRecordDetails(true);
    header.setRunAsCurrentUser(true);

    binding.setHeader(new SforceServiceLocator().getserviceName().getNamespaceURI(),
"DuplicateRuleHeader", header);
```

SEE ALSO:

[DuplicateResult](#)

[DuplicateRule](#)

EmailHeader

The Salesforce user interface allows you to specify whether to send an email when these events occur:

- Create a [Case](#)
- Create a [CaseComment](#)
- Convert [Case](#) email to a [Contact](#)
- Send a New [User](#) email notification
- Make a `resetPassword()` call

In API versions 8.0 and later, you can also send an API request that sends email.

A group event is an [Event](#) for which `IsGroupEvent` is true. The [EventRelation](#) object tracks the users, leads, or contacts that are invited to a group event. Note the following behaviors for group event email sent through the API:

- Sending a group event invitation to a [User](#) respects the `triggerUserEmail` option
- Sending a group event invitation to a [Lead](#) or [Contact](#) respects the `triggerOtherEmail` option

- Email sent when updating or deleting a group event also respect `triggerUserEmail` and `triggerOtherEmail`, as appropriate

API Calls

`create()`, `delete()`, `resetPassword()`, `update()`, `upsert()`

Fields

Element Name	Type	Description
<code>triggerAutoResponseEmail</code>	<code>boolean</code>	Indicates whether to trigger auto-response rules (<code>true</code>) or not (<code>false</code>), for leads and cases. In the Salesforce user interface, this email can be automatically triggered by a number of events, for example creating a case or resetting a user password. If this value is set to <code>true</code> , when a <code>Case</code> is created, if there is an email address for the contact specified in <code>ContactId</code> , the email is sent to that address. If not, the email is sent to the address specified in <code>SuppliedEmail</code> .
<code>triggerOtherEmail</code>	<code>boolean</code>	Indicates whether to trigger email outside the organization (<code>true</code>) or not (<code>false</code>). In the Salesforce user interface, this email can be automatically triggered by creating, editing, or deleting a contact for a case.
<code>triggerUserEmail</code>	<code>boolean</code>	Indicates whether to trigger email that is sent to users in the organization (<code>true</code>) or not (<code>false</code>). In the Salesforce user interface, this email can be automatically triggered by a number of events; resetting a password, creating a new user, or adding comments to a case.

Sample Code—Java

This sample shows how to use the `EmailHeader`. It sets the `triggerAutoResponseEmail` email header field to `true`, which triggers an email to be sent when a case is created. Next, it creates a case. This sample assumes an auto-response rule has been set for cases, and an email address is specified in the contact referenced by `ContactId`.

```
public void createCaseWithAutoResponse(String contactId) {
    try {
        connection.setEmailHeader(true, false, false);
        Case c = new Case();
        c.setSubject("Sample Subject");
        c.setContactId(contactId);
        SaveResult[] sr = connection.create(new SObject[] { c });
        // Parse sr array to see if case was created successfully.
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

LimitInfoHeader

A response header returned from calls to the SOAP API. This header returns limit information for the organization. Use this header to monitor your API limits as you make calls against the organization.

API Calls

All calls, except for [login\(\)](#).

Fields

Element Name	Type	Description
current	string	The number of calls for the specified limit type that have already been used in the organization.
limit	string	The organization's limit for the specified limit type.
type	string	<p>The type of limit information specified in the header.</p> <ul style="list-style-type: none"> • <code>API REQUESTS</code>—contains limit information about API calls for the organization. • <code>API REQUESTS PER APP</code>—contains limit quota information, if there is any, for the currently connected app. API limit app quotas are currently available through a pilot program. For information on enabling this feature for your organization, contact Salesforce. This example includes the limit quota for a sample-connected-app connected app.

```
<limitInfo>
    <appName>sample-connected-app</appName>
    <current>20</current>
    <limit>250</limit>
    <type>API REQUESTS PER APP</type>
</limitInfo>
```

Sample Code

This is an example of a response to a SOAP request for a Merchandise record. The `LimitInfoHeader` contains the API usage information for the organization.

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns="urn:partner.soap.sforce.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:sf="urn:sobject.partner.soap.sforce.com">
    <soapenv:Header>
        <LimitInfoHeader>
            <limitInfo>
                <current>5</current>
                <limit>5000</limit>
            </limitInfo>
        </LimitInfoHeader>
    </soapenv:Header>
    <Body>
        <GetMerchandiseRequest>
            <id>00U200000000000</id>
        </GetMerchandiseRequest>
    </Body>
</soapenv:Envelope>
```

```

        <type>API REQUESTS</type>
    </limitInfo>
</LimitInfoHeader>
</soapenv:Header>
<soapenv:Body>
<queryResponse>
<result xsi:type="QueryResult">
<done>true</done>
<queryLocator xsi:nil="true"/>
<records xsi:type="sf:sObject">
<sf:type>dev_ns__Merchandise__c</sf:type>
<sf:Id>a00D0000008pQSNIA2</sf:Id>
<sf:dev_ns__Description__c>Phone Case for iPhone
    4/4S</sf:dev_ns__Description__c>
<sf:dev_ns__Price__c>16.99</sf:dev_ns__Price__c>
<sf:dev_ns__Stock_Price__c>12.99</sf:dev_ns__Stock_Price__c>
<sf:dev_ns__Total_Inventory__c>108.0</sf:dev_ns__Total_Inventory__c>
<sf:Id>a00D0000008pQSNIA2</sf:Id>
</records>
<size>1</size>
</result>
</queryResponse>
</soapenv:Body>
</soapenv:Envelope>

```

LocaleOptions

Specifies the language of the labels returned.

API Calls

```
describeSObject(), describeSObjects(), describeDataCategoryGroups(),
describeDataCategoryGroupStructures()
```

Fields

Element Name	Type	Description
language	string	Specifies the language of the labels returned. The value must be a valid user locale (language and country), such as <code>de_DE</code> or <code>en_GB</code> . For more information on locales, see the Language field on the <code>CategoryNodeLocalization</code> object.

Sample Code—Java

This sample sets the `LocaleOptions` header to the locale of the logged-in user, and then performs a describe on Account.

```
public void localeOptionsExample() {
    try {
        connection.setLocaleOptions("en_US");
```

```

        connection.describeSObject("Account");
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}

```

LoginScopeHeader

Specifies your organization ID so that you can authenticate Self-Service users for your organization using the existing [login\(\)](#).

API Calls

[login\(\)](#)

Fields

Element Name	Type	Description
organizationId	ID	The ID of the organization against which you will authenticate Self-Service users.
portalId	ID	Specify only if user is a Customer Portal user. The ID of the portal for this organization. The ID is available in the Salesforce user interface: <ul style="list-style-type: none"> From Setup, enter <i>Customer Portal Settings</i> in the Quick Find box, then select Customer Portal Settings Select a Customer Portal name, and on the Customer Portal detail page, the URL of the Customer Portal displays. The Portal ID is in the URL.

Sample Code—C#

This sample shows how to use the `LoginScopeHeader`. It sets the organization ID and the portal ID for a Customer Portal user. It also sets the `CallOptions` header. It then logs the specified user in.

```

/// Demonstrates how to set the LoginScopeHeader values.
public void LoginScopeHeaderSample()
{
    // Web Reference to the imported Partner WSDL.
    APISamples.partner.SforceService partnerBinding;

    string username = "USERNAME";
    string password = "PASSWORD";

    // The real Client ID will be an API Token provided by salesforce.com
    // to partner applications following a security review. For more details,
    // see the Security Review FAQ in the online help.
    string clientId = "SampleCaseSensitiveToken/100";

    partnerBinding = new SforceService();
    partnerBinding.CallOptionsValue = new CallOptions();

```

```

partnerBinding.CallOptionsValue.client = clientId;

// To authenticate Self-Service users, we need to set the OrganizationId
// in the LoginScopeHeader.
string orgId = "00ID0000OrgFoo";
partnerBinding.LoginScopeHeaderValue = new LoginScopeHeader();
partnerBinding.LoginScopeHeaderValue.organizationId = orgId;
// Specify the Portal ID if the user is a Customer Portal user.
string portalId = "00ID0000FooPtl";
partnerBinding.LoginScopeHeaderValue.portalId = portalId;

try
{
    APISamples.partner.LoginResult lr =
        partnerBinding.login(username, password);
}
catch (SoapException e)
{
    Console.WriteLine(e.Code);
    Console.WriteLine(e.Message);
}
}
}

```

MrUHeader

In API version 7.0 and later, the `create()`, `update()`, and `upsert()` calls do not update the list of most recently used (MRU) items in the Recent Items section of the sidebar in the Salesforce user interface unless this header is used. Be advised that using this header to update the Recent Items list may negatively impact performance.

API Calls

`create()`, `merge()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Element Name	Type	Description
updateMru	boolean	<p>Indicates whether to update the list of most recently used items (<code>true</code>) or not (<code>false</code>).</p> <p>For <code>retrieve()</code>, if the result has only one row, the MRU is updated to the ID of the retrieve result.</p> <p>For <code>query()</code>, if the result has only one row and the ID field is selected, the MRU is updated to the ID of the query result.</p>

Sample Code—Java

This sample turns on the MRU list update option by setting the `MruHeader` to `true`. Next, it creates an account.

```
public void mruHeaderSample() {
    connection.setMruHeader(true);
    Account account = new Account();
    account.setName("This will be in the MRU");
    try {
        SaveResult[] sr = connection.create(new SObject[]{account});
        System.out.println("ID of account added to MRU: " +
            sr[0].getId());
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

OwnerChangeOptions

Represents actions that can be performed when a record's owner is changed. Available with these options in API version 35.0 and later.

API Calls

`update()`, `upsert()`

Fields

Element Name	Type	Description
options	<code>OwnerChangeOption[]</code>	Represents a flag for a specific action performed when changing a record owner through an update or upsert call.

OwnerChangeOption Fields

Element Name	Type	Description
execute	<code>boolean</code>	If <code>true</code> , the action represented by the type field is performed. If <code>false</code> , the action represented by the type field is skipped.
type	enum of a string	Represents the action performed or skipped, according to the given value for the execute field, when changing a record owner during an update or upsert call. The following types can be used.

EnforceNewOwnerHasReadAccess

If true, the record's new owner must have at least read access on the record. Available in API version 36.0 and later.

Element Name	Type	Description
		TransferOpenActivities
		If true, the record's open activities are transferred to the new owner
		TransferNotesAndAttachments
		If true, the record's notes, attachments, and Google Docs are transferred to the new record owner. If false, the original record owner retains ownership.
		TransferOtherOpenOpportunities
		If true and the record is an account, open opportunities associated with the account and not owned by the current record owner are transferred to the new owner.
		TransferOwnedOpenOpportunities must be set to execute when executing this option.
		TransferOwnedOpenOpportunities
		If true and the record is an account, open opportunities associated with the account and owned by the account owner are transferred to the new owner.
		TransferContracts
		If true and the record is an account, contracts associated with the account and owned by the account owner are transferred to the new owner.
		TransferOrders
		If true and the record is an account, the draft standalone orders associated with the account and draft orders associated with transferred contracts owned by the account owner are transferred to the new owner.
		TransferContacts
		If true and the record is a business account, contacts associated with the account are transferred to the new owner.

Sample Code—Java

This sample creates an account, a note, an opportunity, and task for the account, and sets the owner change options so that the note, opportunity and task are transferred to the new owner along with account.

```
public void ownerChangeOptionsHeaderSample() {  
  
    // Create account. Accounts don't transfer activities, notes, or attachments by default  
  
    Account account = new Account();  
    account.setName("Account");  
    com.sforce.soap.enterprise.SaveResult[] sr = connection.create(new  
com.sforce.soap.enterprise.sobject.SObject[] { account } );  
    String accountId = null;  
  
    if(sr[0].isSuccess()) {  
        System.out.println("Successfully saved the account");  
        accountId = sr[0].getId();  
  
        // Create a note, a task, and an opportunity for the account  
    }  
}
```

```
Note note = new Note();
note.setTitle("Note Title");
note.setBody("Note Body");
note.setParentId(accountId);

Task task = new Task();
task.setWhatId(accountId);

Opportunity opportunity = new Opportunity();
opportunity.setName("Opportunity");
opportunity.setStageName("Prospecting");
Calendar dt = connection.getServerTimestamp().getTimestamp();
dt.add(Calendar.DAY_OF_MONTH, 7);
opportunity.setCloseDate(dt);
opportunity.setAccountId(accountId);

sr = connection.create(new com.sforce.soap.enterprise.SObject[] { note,
task, opportunity } );

if(sr[0].isSuccess()) {
    System.out.println("Successfully saved the note, task, and opportunity");

    com.sforce.soap.enterprise.QueryResult qr = connection.query("SELECT Id FROM
User WHERE FirstName = 'Jane' AND LastName = 'Doe'");
    String newOwnerId = qr.getRecords()[0].getId();
    account.setId(accountId);
    account.setOwnerId(newOwnerId);

    // Set owner change options so account's child note, task, and opportunity
transfer to new owner
    OwnerChangeOption opt1 = new OwnerChangeOption();
    opt1.setExecute(true);
    opt1.setType(OwnerChangeOptionType.TransferOwnedOpenOpportunities); // Transfer
Open opportunities owned by the account's owner

    OwnerChangeOption opt2 = new OwnerChangeOption();
    opt2.setExecute(true);
    opt2.setType(OwnerChangeOptionType.TransferOpenActivities);

    OwnerChangeOption opt3 = new OwnerChangeOption();
    opt3.setExecute(true);
    opt3.setType(OwnerChangeOptionType.TransferNotesAndAttachments);

    connection.setOwnerChangeOptions(new OwnerChangeOption[] {opt1, opt2, opt3});
    connection.update(new com.sforce.soap.enterprise.SObject[] { account })
};

    // The account's note, task, and opportunity should be transferred to the new
owner.
}

} else {
```

```

        System.out.println("Account save failed: " + sr[0].getErrors().toString());
    }
}

```

PackageVersionHeader

Specifies the package version for each installed managed package.

A managed package can have several versions with different content and behavior. This header allows you to specify the version used for each package referenced by your API client.

If a package version is not specified, the API client uses the version of the package specified in Setup (enter *API* in the Quick Find box, then select **API**).

This header is available in API version 16.0 and later.

Associated API Calls

```
convertLead(), create(), delete(), describeGlobal(), describeLayout(), describeSObject(),
describeSObjects(), describeSoftphoneLayout(), describeTabs(), merge(), process(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Element Name	Type	Description
packageVersions	PackageVersion[]	A list of package versions for installed managed packages referenced by your API client.

PackageVersion

Specifies a version of an installed managed package. A package version is *majorNumber.minorNumber*, for example 2.1.

Fields

Field	Type	Description
majorNumber	int	The major version number of a package version.
minorNumber	int	The minor version number of a package version.
namespace	string	The unique namespace of the managed package.

Sample Code—Java

This sample sets the package version for one installed package in the `PackageVersionHeader`. Next, it executes the code passed into this method via the `executeAnonymous` Apex method.

```

public void PackageVersionHeaderSample(String code) throws Exception
{
    _PackageVersionHeader pvh = new _PackageVersionHeader();

```

```

        PackageVersion pv = new PackageVersion();
        pv.setNamespace("installedPackageNamespaceHere");
        pv.setMajorNumber(1);
        pv.setMinorNumber(0);
        // In this case, we are only referencing one installed package.
        PackageVersion[] pvs = new PackageVersion[]{pv};
        pvh.setPackageVersions(pvs);

        apexBinding.setHeader(new SforceServiceLocator().getServiceName().getNamespaceURI(),
            "PackageVersionHeader", pvh);
        // Execute the code passed into the method.
        ExecuteAnonymousResult r = apexBinding.executeAnonymous(code);
        if (r.isSuccess()) {
            System.out.println("Code executed successfully");
        }
        else {
            System.out.println("Exception message: " + r.getExceptionMessage());
            System.out.println("Exception stack trace: " + r.getExceptionStackTrace());
        }
    }
}

```

QueryOptions

Specifies the preferred batch size for queries. The system sometimes creates batches that are larger or smaller than the specified size to maximize performance.

Associated API Calls

[query\(\)](#), [queryMore\(\)](#), [retrieve\(\)](#)

Fields

Element Name	Type	Description
batchSize	int	<p>The batch size for the number of records returned in a query() or queryMore() call. Child objects count toward the number of records for the batch size. For example, in relationship queries, multiple child objects are returned per parent row returned.</p> <p>The default is 500; the minimum is 200, and the maximum is 2,000. There is no guarantee that the requested batch size requested is the actual batch size; changes are sometimes made to maximize performance.</p>

Sample Code

For code examples, see Change the Batch Size in Queries in the [Salesforce SOQL and SOSL Reference Guide](#).

SessionHeader

Specifies the session ID returned from the login server after a successful [login \(\)](#). This session ID is used in all subsequent calls.

In version 12.0 and later, include the API namespace in the SOAP message associated with this header. The namespace is defined in the enterprise or partner WSDL.

API Calls

All calls, including utility calls.

Fields

Element Name	Type	Description
sessionId	string	Session ID returned by the login () call to be used for subsequent call authentication.

Sample Code

See the examples provided for [login \(\)](#).

UserTerritoryDeleteHeader

Specify a user to whom open opportunities are assigned when the current owner is removed from a territory. If this header is not used or the value of its element is null, the opportunities are transferred to the forecast manager in the territory above, if one exists. If one does not exist, the user being removed from the territory keeps the opportunities.

API Calls

[delete \(\)](#)

Fields

Element Name	Type	Description
transferToUserId	ID	The ID of the user to whom open opportunities in that user's territory will be assigned when an opportunity's owner (user) is removed from a territory.

SEE ALSO:

[Opportunity Forecast Override Business Rules](#)

USING THE API WITH SALESFORCE FEATURES

CHAPTER 13 Implementation Considerations

In this chapter ...

- Choosing a User for an Integration
- Login Server URL
- Log in to the Login Server
- Typical API Call Sequence
- Salesforce Sandbox
- Multiple Instances of Salesforce Database Servers
- Content Type Requirement
- Monitoring API Traffic
- API Usage Metering
- Compression
- HTTP Persistent Connections
- HTTP Chunking
- Internationalization and Character Sets
- XML Compliance
- .NET, Non-String Fields, and the Enterprise WSDL

Before you build an integration application or other client application, consider the data management, use limits, and communication issues explained in this section.

Choosing a User for an Integration

When your client application connects to the API, it must first log in. You must specify a user to log in to Salesforce when calling `login()`. Client applications run with the permissions and sharing of the logged-in user. The following sections include information that help you to decide how to configure a user for your client application.

Permissions

An organization's Salesforce administrator controls the availability of various features and views by configuring profiles and permission sets, and assigning users to them. To access the API (to issue calls and receive the call results), a user must be granted the "API Enabled" permission. Client applications can query or update only those objects and fields to which they have access via the permissions of the logged-in user.

If the client application logs in as a user who has access to data via a sharing rule, then the API must issue an extra query to check access. To avoid this, log in as a user with the "Modify All Data" permission. This can speed up the call response time. If providing the "Modify All Data" permission is too permissive for a particular user, consider using the "Modify All" object-level permission to restrict data access on an object basis. For more information on permissions, see [Factors that Affect Data Access](#).

Limits

There is a limit on the number of queries that a user can execute concurrently. A user can have up to 10 query cursors open at a time. If 10 `QueryLocator` cursors are open when a client application, logged in as the same user, attempts to open a new one, then the oldest of the 10 cursors is released. If the client application attempts to open the released query cursor, an error results.

Multiple client applications can log in using the same `username` argument. However, this approach increases your risk of getting errors due to query limits.

If multiple client applications are logged in using the same user, they all share the same session. If one of the client applications calls `logout()`, it invalidates the session for all the client applications. Using a different user for each client application makes it easier to avoid these limits.



Note: In addition to user limits, there are limits for API requests for each organization. For more information, see [API Usage Metering](#).

Login Server URL

The SOAP implementation of the API also provides a single login server. You can log in to any organization via a single entry point, without having to hard-code the instance for your organization. To access an organization via the API, you must first authenticate the session by sending a `login()` request to the login server at one of the following URLs, depending on your choice of WSDL.

- `https://login.salesforce.com/services/Soap/c/38.0` (enterprise WSDL)
- `https://login.salesforce.com/services/Soap/u/38.0` (partner WSDL)

The less secure version of each URL is also supported.

`http://login.salesforce.com/services/Soap/c/38.0`

- `http://login.salesforce.com/services/Soap/c/38.0` (enterprise WSDL)
- `http://login.salesforce.com/services/Soap/u/38.0` (partner WSDL)

The less secure version of the URL is supported, but not recommended. It is helpful for debugging through proxy servers.

All subsequent calls to the server during the session should be made to the URL returned in the [login\(\)](#) response, which points to the server instance for your organization.

Log in to the Login Server

Before invoking any other calls, a client application must first invoke the [login\(\)](#) call to establish a session with the login server, set the returned server URL as the target server for subsequent API requests, and set the returned session ID in the SOAP header to provide server authorization for subsequent API requests. For more information, see [login\(\)](#) and [Step 4: Walk Through the Sample Code](#).

Salesforce checks the IP address from which the client application is logging in, and blocks logins from unknown IP addresses. For a blocked login via the API, Salesforce returns a login fault. Then, the user must add their security token to the end of their password in order to log in. A security token is an automatically-generated key from Salesforce. For example, if a user's password is *mypassword*, and their security token is *XXXXXXXXXXXX*, then the user must enter *mypasswordXXXXXXXXXX* to log in. Users can obtain their security token by changing their password or resetting their security token via the Salesforce user interface. When a user changes their password or resets their security token, Salesforce sends a new security token to the email address on the user's Salesforce record. The security token is valid until a user resets their security token, changes their password, or has their password reset. When the security token is invalid, the user must repeat the login process to log in. To avoid this, the administrator can make sure the client's IP address is added to the organization's list of trusted IP addresses. For more information, see [Security Token](#).

Once you are logged in, you can issue API calls. For each operation, client applications submit a synchronous request to the API, await the response, and process the results. The API commits any changed data automatically.

API calls:

- [Core Calls](#)
- [Describe Calls](#)
- [Utility Calls](#)

Typical API Call Sequence

For each call, your client application typically:

1. Prepares the request by defining request parameters, if applicable.
2. Invokes the call, which passes the request with its parameters to the Force.com Web Service for processing.
3. Receives the response from the API.
4. Handles the response, either by processing the returned data (for a successful invocation) or by handling the error (for a failed invocation).

Salesforce Sandbox

Professional, Enterprise, Unlimited, and Performance Edition customers have access to the Salesforce Sandbox, which is a testing environment that offers a full or partial copy of your Salesforce org's live production data. For more information about Salesforce Sandbox, visit the Salesforce.com Community website at www.salesforce.com/community or see "Sandboxes" in the Salesforce Help.

To access your org's Salesforce Sandbox via the API, use the following URLs to make login requests.

- <https://test.salesforce.com/services/Soap/c/38.0> (enterprise WSDL)
- <https://test.salesforce.com/services/Soap/u/38.0> (partner WSDL)

Multiple Instances of Salesforce Database Servers

Salesforce provides many database server instances. Although orgs are generally allocated by geographic regions, an org may be on any instance.

Content Type Requirement

In the API version 7.0 and later, all requests must contain a correct content type HTTP header, for example: `Content-Type: text/xml; charset=utf-8`. Earlier versions of the API do not enforce this requirement.

Monitoring API Traffic

You can monitor the number of API requests generated by your org in two ways.

- Any user can see the number of API requests sent in the last 24 hours. To view the information, from Setup, enter *Company Information* in the Quick Find box, then select **Company Information**. Look for the “API Requests, Last 24 Hours” field in the right column.
- If a user has the “Modify All Data” permission, the user can view a report of the API requests sent for the last seven days. To view the information, click the Reports tab, scroll to the **Administrative Reports** section and click the **API Usage Last 7 Days** link. Users can sort the report by any of the fields listed in the **Summarize Information by:** drop-down list. For more information about sorting, filtering, or customizing reports, see the Salesforce online help for reports.

API Usage Metering

To maintain optimum performance and ensure that the Force.com API is available to all our customers, Salesforce balances transaction loads by imposing two types of limits:

- Concurrent API Request Limits
- Total API Request Limits

When a call exceeds a request limit, an error is returned.

Concurrent API Request Limits

The following table lists the limits for various types of organizations for concurrent requests (calls) with a duration of 20 seconds or longer:

Org Type	Limit
Developer Edition	5
Trial orgs	5
Production orgs	25
Sandbox	25

Total API Request Limits

The following table lists the limits for the total API requests (calls) per 24-hour period for an org.

Salesforce Edition	API Calls Per License Type	Total Calls Per 24-Hour Period
Developer Edition	N/A	15,000
• Enterprise Edition • Professional Edition with API access enabled	• Salesforce: 1,000 • Salesforce Platform: 1,000 • Force.com - One App: 200 • Gold Partner: 200	15,000 + (number of licenses X calls per license type), up to a maximum of 1,000,000
• Unlimited Edition • Performance Edition	• Salesforce: 5,000 • Salesforce Platform: 5,000 • Force.com - One App: 200 • Gold Partner: 200	15,000 + (number of licenses X calls per license type)
Sandbox	N/A	5,000,000

Limits are enforced against the aggregate of all API calls made by the org in a 24 hour period. Limits are not on a per-user basis. When an org exceeds a limit, all users in the org can be temporarily blocked from making additional calls. Calls are blocked until usage for the preceding 24 hours drops below the limit.

 **Note:** Limits are automatically enforced for all editions.

Any action that sends a call to the API counts toward usage limits, except the following:

- Outbound messages
- Apex callouts

You can create an API usage metering notification, so that Salesforce will send an email to a designated user when API limits have exceeded a specified limit in a specified time period. For more information, see "API Usage Notifications" in the Salesforce online help.

There are also limits on the number of requests allowed per org from the Salesforce user interface. For details, see "Concurrent Usage Limits" in the Salesforce online help.

Example API Usage Metering Calculations

The following examples illustrate API usage metering calculations for several scenarios:

- For an Enterprise Edition organization with fifteen Salesforce licenses, the request limit is 15,000 requests (15 licenses X 1,000 calls).
- For an Enterprise Edition organization with 15,000 Salesforce licenses, the request limit is 1,000,000 (the number of licenses X 1,000 calls is greater than the maximum value, so the lower limit of 1,000,000 is used).
- For a Developer Edition organization that made 14,500 calls at 5:00 AM Wednesday, 499 calls at 11:00 PM Wednesday, only one more call could successfully be made until 5:00 AM Thursday.
- For an Enterprise Edition organization with 60 Gold Partner licenses, the request limit is 15,000 (the number of licenses X 200 calls is less than the minimum value of 15,000).

Increasing Total API Request Limit

The calculation of the API request limit based on user licenses is designed to allow sufficient capacity for your organization based on your number of users. If you need a higher limit and you don't want to purchase additional user licenses or upgrade to Performance Edition, you can purchase additional API calls. For more information, contact your account representative.

Before you purchase additional API calls, you should perform a due diligence of your current API usage. Client applications, whether it is your own enterprise applications or partner applications, that make calls to the API can often be optimized to use fewer API calls to do the same work. If you are using a partner product, you should consult with the vendor to verify that the product is making optimal use of the API. A product that makes inefficient use of the API will incur unnecessary cost for your organization.

Compression

The API allows the use of compression on the request and the response, using the standards defined by the HTTP 1.1 specification. This is automatically supported by some SOAP/WSDL clients, and can be manually added to others. Visit

<https://developer.salesforce.com/page/Tools> for more information on particular clients.

Compression is not used unless the client specifically indicates that it supports compression. For better performance, we suggest that clients accept and support compression as defined by the HTTP 1.1 specification.

To indicate that the client supports compression, you should include the HTTP header "Accept-Encoding: gzip, deflate" or a similar heading. The API compresses the response if the client properly specifies this header. The response includes the header "Content-Encoding: deflate" or "Content-Encoding: gzip," as appropriate. You can also compress any request by including a "Content-Encoding: deflate" or "gzip" header.

Most clients are partially constrained by their network connection, even on a corporate LAN. The API allows the use of compression to improve performance. Almost all clients can benefit from response compression, and many clients may benefit from compression of requests as well. The API supports deflate and gzip compression according the HTTP 1.1 specification.

Response Compression

The API can optionally compress responses. Responses are compressed only if the client sends an Accept-Encoding header with either gzip or deflate compression specified. The API is not required to compress the response even if you have specified Accept-Encoding, but it normally does. If the API compresses the response, it also specifies a Content-Encoding header with the name of the compression algorithm used, either gzip or deflate.

Request Compression

Clients can also compress requests. The API decompresses any requests before processing. The client must send up a Content-Encoding HTTP header with the name of the appropriate compression algorithm. For more information, see:

- Content-Encoding at: www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.11
- Accept-Encoding at: www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.3
- Content Codings at: www.w3.org/Protocols/rfc2616/rfc2616-sec3.html#sec3.5



Note: To implement request SOAP compression in a Java client with WSC (Web Service Connector), call `setCompression()` on the `Config` you use to instantiate a `Connection` object with. For an example, see `login()` sample on page 2185 code.

HTTP Persistent Connections

Most clients achieve better performance if they use HTTP 1.1 persistent connection to reuse the socket connection for multiple requests. Persistent connections are normally handled by your SOAP/WSDL client automatically. For more details, see the HTTP 1.1 specification at:

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec8.html#sec8.1>

HTTP Chunking

Clients that use HTTP 1.1 may receive chunked responses. Chunking is normally handled by your SOAP/WSDL client automatically.

Internationalization and Character Sets

The API supports either full Unicode characters or ISO-8859-1 characters. The character set for your organization depends on the Salesforce instance your organization uses. If your organization logs into `ssl.salesforce.com`, then your encoding is `ISO-8859-1`. All other instances use `UTF-8`. You can determine the character set for your organization by calling `describeGlobal()` and inspecting the `encoding` value returned in the `DescribeGlobalResult`.

If your organization uses ISO-8859-1 encoding, then all data sent to the API must be encoded in ISO-8859-1. Characters outside the valid ISO-8859-1 range might be truncated or cause an error.

 **Note:** The API response is encoded in the character set used by your organization (UTF-8 or ISO-8859-1). Either way, the encoded data is usually handled for you by the SOAP client.

XML Compliance

The API is based on XML, which requires all documents to be well formed. Part of that requirement is that certain Unicode characters are not allowed in an XML document, even in an escaped form, and that others must be encoded according to their location. Normally this is handled for you by any standard SOAP or XML client. Clients must be able to parse any normal XML escape sequence, and must not pass up invalid XML characters.

Some characters, as mentioned, are illegal even if they are escaped. The illegal characters include unpaired Unicode surrogates and a few other Unicode characters. All are seldom-used control characters that are usually not important in any data, and tend to cause problems with many programs. Although they are not allowed in XML documents, they are allowed in HTML documents and may be present in Salesforce data. The illegal characters will be stripped from any API response.

Illegal characters:

- `0xFFFFE`
- `0xFFFFF`
- Control characters `0x0` - `0x19`, except the following characters, which are legal: `0x9`, `0xA`, `0xD`, tab, newline, and carriage return)
- `0xD800` - `0xDFFF`, unless they're used to form a surrogate pair

For UTF-8 encoding, Salesforce supports only the basic UCS-2 encoding (two byte, Basic Multilingual Plane), and does not support any of the extended UCS-4 characters. UCS-4 support is extremely rare in any system. UCS-2 is the encoding that Java and Windows support. For more information about XML characters and character sets, see: <http://www.w3.org/TR/REC-xml#charsets>.

.NET, Non-String Fields, and the Enterprise WSDL

If you use .NET with the enterprise WSDL, .NET will generate an extra Boolean field for each non-string field. For example, if you had a date value in `MyField__c`, .NET generates a Boolean field for each. In this example, the generated fields would be `MyField__cSpecified` and `LastModifiedSpecified`. These field values are `false` by default. If a Specified field value is `false`, then the values in the corresponding original field will not be included in the SOAP message.

For example, before the values in the currency field `annualRevenue` can be included in a SOAP message generated by your client application, the value of `annualRevenueSpecified` must be set to `true`.

```
account.annualRevenue = 10000;  
account.annualRevenueSpecified = true;
```

CHAPTER 14 Outbound Messaging

In this chapter ...

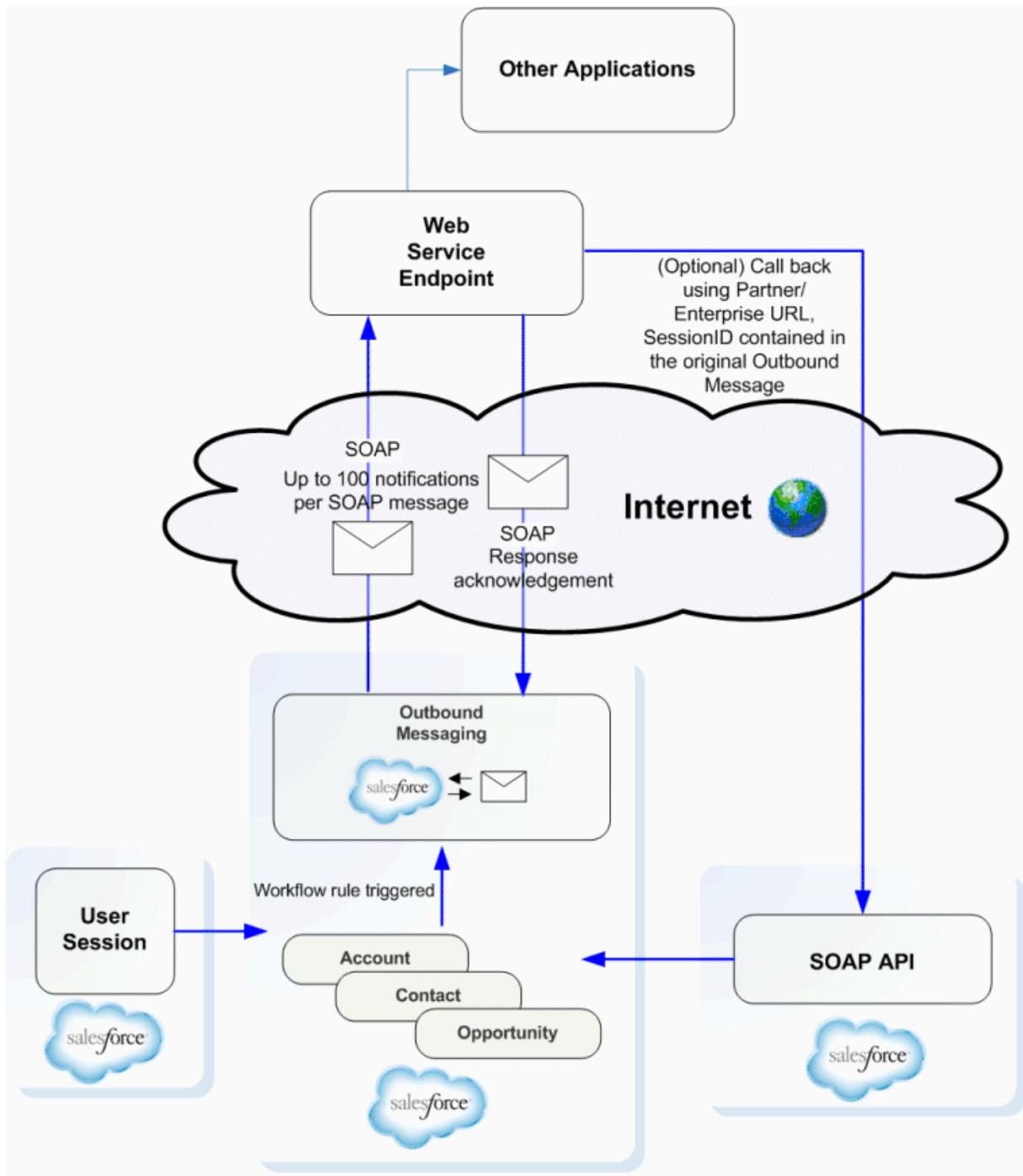
- Understanding Outbound Messaging
- Understanding Notifications
- Setting Up Outbound Messaging
- Considerations for Security
- Understanding the Outbound Messaging WSDL
- Building a Listener

Outbound messaging allows you to specify that changes to fields within Salesforce can cause messages with field values to be sent to designated external servers.

Outbound messaging is part of the workflow rule functionality in Salesforce. Workflow rules watch for specific kinds of field changes and trigger automatic Salesforce actions, such as sending email alerts, creating task records, or sending an outbound message.

Understanding Outbound Messaging

Outbound messaging uses the `notifications()` call to send SOAP messages over HTTP(S) to a designated endpoint when triggered by a workflow rule.



After you set up outbound messaging, when a triggering event occurs, a message is sent to the specified endpoint URL. The message contains the fields specified when you created the outbound message. Once the endpoint URL receives the message, it can take the information from the message and process it. To do that, you need to examine the outbound messaging WSDL.

Understanding Notifications

A single SOAP message can include up to 100 notifications. Each notification contains the object ID and a reference to the associated [sObject](#) data. Note that if the information in the object changes after the notification is queued but before it is sent, only the updated information will be delivered.

If you issue multiple discrete calls, the calls may be batched together into one or more SOAP messages.

Messages will be queued locally. A separate background process performs the actual sending, to preserve message reliability:

- If the endpoint is unavailable, messages will stay in the queue until sent successfully, or until they are 24 hours old. After 24 hours, messages are dropped from the queue.
- If a message cannot be delivered, the interval between retries increases exponentially, up to a maximum of two hours between retries.
- Messages are retried independent of their order in the queue. This may result in messages being delivered out of order.
- You cannot build an audit trail using outbound messaging. While each message should be delivered at least once, it may be delivered more than once. Also, it may not be delivered at all if delivery cannot be done within 24 hours. Finally, as noted above, the source object may change after a notification is queued but before it is sent, so the endpoint will only receive the latest data, not any intermediate changes.
- Because a message may be delivered more than once, your listener client should check the notification IDs delivered in the notification before processing.

 **Note:** Instead of polling, which was required in previous releases, you can now use outbound messaging to trigger execution logic when Salesforce raises an event. In previous versions of the API, client applications had to poll Salesforce to find out if relevant changes had occurred. Because most changes eventually trigger workflow if a rule exists for it, you can use this to trigger actions based on Salesforce events.

The metadata needed for outbound messaging, including the definition of the [notifications \(\)](#) call, which sends the outbound SOAP message to an external service, is in a separate WSDL. The WSDL is created and available from the Salesforce user interface once a workflow rule has been associated with an outbound message. The WSDL is bound to the outbound message and contains the instructions about how to reach the endpoint service and what data is sent to it. For more information about setting up outbound messaging, see [Defining Outbound Messaging](#).

Setting Up Outbound Messaging

Before you can use outbound messaging, you must set it up via the Salesforce user interface.

- [Setting Up User Profiles](#)
- [Defining Outbound Messaging](#)
- [Downloading the Salesforce Client Certificate](#)
- [Viewing Outbound Messages](#)
- [Tracking Outbound Message Status](#)

Setting Up User Profiles

It's possible to create circular changes with outbound messaging. For example, if a user is performing integrations that trigger workflow, and the workflow actions trigger account updates, those account updates trigger new workflow, and so on. To prevent these circular changes, you can disable a user's ability to send outbound messages.

Here's another example of a circular change scenario.

1. You configure an outbound message to include a `sessionId` and specify a user in the **User to send as** field. The user doesn't have outbound messaging disabled.
2. A change in a contact record triggers an outbound message from the specified user, with the `sessionId` to your outbound message listener.
3. Your outbound message listener calls the Force.com API and updates the same contact record which triggered the outbound message.
4. The update triggers an outbound message.
5. Your outbound message listener updates the record.
6. The update triggers an outbound message.
7. Your outbound message listener updates the record.

To disable outbound message notifications for a user, deselect **Send Outbound Messages** in the user's [Profile](#). We recommend specifying a single user to respond to outbound messages and disabling this user's ability to send outbound messages.

Defining Outbound Messaging

To define outbound messages, use this procedure in the Salesforce user interface:

1. From Setup, enter *Outbound Messages* in the Quick Find box, then select **Outbound Messages**.
2. Click **New Outbound Message**.
3. Choose the object that has the information you want included in the outbound message, and click **Next**.
4. Configure the outbound message.
 - a. Enter a name and description for this outbound message.
 - b. Enter an endpoint URL for the recipient of the message. Salesforce sends a SOAP message to this endpoint.

For security reasons, Salesforce restricts the outbound ports you may specify to one of the following:

- 80: This port only accepts HTTP connections.
 - 443: This port only accepts HTTPS connections.
 - 1024–66535 (inclusive): These ports accept HTTP or HTTPS connections.
- c. Select the Salesforce user to use when sending the message by specifying a username in the **User to send as** field. The chosen user controls data visibility for the message that is sent to the endpoint.
 - d. Select **Include Session ID** if you want a `sessionId` to be included in the outbound message. Include the `sessionId` in your message if you intend to make API calls back to Salesforce from your listener. The `sessionId` represents the user defined in the previous step and not the user who triggered the workflow.
 - e. Select the fields you want included in the outbound message and click **Add**.

5. Click **Save**, and review the outbound message detail page:

- The **API Version** field is automatically generated and set to the current API version when the outbound message was created. This API version is used in API calls back to Salesforce using the enterprise or partner WSDL. The **API Version** can only be modified by using the Metadata API.
- Click **Click for WSDL** to view the WSDL associated with this message.

The WSDL is bound to the outbound message and contains the instructions about how to reach the endpoint service and what data is sent to it.



Note: If you don't have these options, your org doesn't have outbound messaging enabled. Contact Salesforce to enable outbound messaging for your org.

Downloading the Salesforce Client Certificate

Your application (endpoint) server's SSL/TLS may be configured to require client certificates (two-way SSL/TLS), in order to validate the identity of the Salesforce server when it takes the role of client to your server. If this is the case, you can download the Salesforce client certificate from the Salesforce application user interface. This is the client certificate that Salesforce sends with each outbound message for authentication.

1. From Setup, enter **API** in the **Quick Find** box, then select **API**.
2. On the API WSDL page, click **Manage API Client Certificate**.
3. On the Certificate and Key Management page, in the API Client Certificate section, click the **API Client Certificate**.
4. On the Certificates page, click **Download Certificate**. The .crt file is saved in the download location specified in your browser.

Import the downloaded certificate into your application server, and configure your application server to request the client certificate. The application server then checks that the certificate used in the SSL/TLS handshake matches the one you downloaded.



Note: Your application (endpoint) server must send any intermediate certificates in the certificate chain, and the certificate chain must be in the correct order. The correct order is:

1. Server certificate
2. Intermediate certificate that signed the server certificate if the server certificate wasn't signed directly by a root certificate
3. Intermediate certificate that signed the certificate in step 2
4. Any remaining intermediate certificates

Don't include the root certificate authority certificate. The root certificate isn't sent by your server. Salesforce already has its own list of trusted certificates on file, and a certificate in the chain must be signed by one of those root certificate authority certificates.

Viewing Outbound Messages

To view existing outbound messages, from Setup, enter **Outbound Messages** in the **Quick Find** box, then select **Outbound Messages** in the Salesforce user interface.

- Click **New Outbound Message** to define a new outbound message.
- Click **View Message Delivery Status** to track the status of an outbound message.
- Select an existing outbound message to view details about it or view workflow rules and approval processes that use it.
- Click **Edit** to make changes to an existing outbound message.
- Click **Del** to delete an outbound message.

Tracking Outbound Message Status

To track the status of an outbound message, from Setup, enter *Outbound Messages* in the Quick Find box, select **Outbound Messages**, and then click **View Message Delivery Status**. You can perform several tasks on this page.

- View the status of your outbound messages, including the total number of attempted deliveries.
- View the action that triggered the outbound message by clicking any workflow or approval process action ID.
- Click **Retry** to change the **Next Attempt** date to now. This action causes the message delivery to be immediately retried.
- Click **Del** to permanently remove the outbound message from the queue.

Considerations for Security

To use outbound messaging, ensure that no third party can send messages to the endpoint while pretending to be from Salesforce:

- Lock down the client application's listener to accept requests only from Salesforce IP ranges. While this action guarantees that the message came from Salesforce, it does not guarantee that another customer is not pointing to your endpoint and sending messages. For an up-to-date list of Salesforce IP ranges, see [What are the Salesforce IP Addresses to whitelist?](#)
- Use SSL/TLS. Using SSL/TLS provides confidentiality while data is transported across the internet. Without it, a malicious third party can eavesdrop on your data. This issue is especially important if you pass data with privacy requirements and you pass a `SessionId` with the message. Also, we authenticate the certificate presented on connection, ensure that it is from a valid Certificate Authority, and check that the domain in the certificate matches the one Salesforce is trying to connect. This prevents us from communicating with the wrong endpoint.
- If the configuration of your application (endpoint) server's SSL/TLS allows, validate the identity of the Salesforce server when it takes the role of a client to your server, using the Salesforce client certificate. For instructions to download the certificate, see [Downloading the Salesforce Client Certificate](#).
- The organization `Id` is included in each message (see [ID Field Type](#) for more information about the `Id` field type). Your client application should validate that messages contain your organization `Id`.

Understanding the Outbound Messaging WSDL

The rest of this topic examines relevant sections of the outbound messaging WSDL. Your WSDL may differ, depending on the choices you made when you set up outbound messaging for a particular event on a particular object.

notifications()

This section defines the `notifications()` call, which creates an outbound message containing specified fields and values for a particular object or objects, and sends the values to a specified endpoint URL:

```
<schema elementFormDefault="qualified" xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://soap.sforce.com/2005/09/outbound">
<import namespace="urn:enterprise.soap.sforce.com" />
<import namespace="urn:sobject.enterprise.soap.sforce.com" />

<element name="notifications">
  <complexType>
    <sequence>
      <element name="OrganizationId" type="ent:ID" />
      <element name="ActionId" type="ent:ID" />
```

```

<element name="SessionId" type="xsd:string" nillable="true" />
<element name="EnterpriseUrl" type="xsd:string" />
<element name="PartnerUrl" type="xsd:string" />
<element name="Notification" maxOccurs="100"
        type="tns:OpportunityNotification" />
    </sequence>
</complexType>
</element>
</schema>

```

Use this table to understand the elements named in the notifications method definition:

Name	Type	Description
OrganizationId	ID	ID of the organization sending the message.
ActionId	string	The workflow rule (action) that triggers the message.
SessionId	string	Optional, a session ID to be used by endpoint URL client that is responding to the outbound message. It is used by the receiving code to make calls back to Salesforce.
EnterpriseURL	string	URL to use to make API calls back to Salesforce using the enterprise WSDL.
PartnerURL	string	URL to use to make API calls back to Salesforce using the partner WSDL.
Notification	Notification	Defined in the next section, contains the object datatype and its Id, for example OpportunityNotification or ContactNotification.

The Notification datatype is defined in the WSDL. In the following example, a Notification for opportunities is defined, based on the Notification entry of the [notifications \(\)](#) call definition:

```

<complexType name="OpportunityNotification">
    <sequence>
        <element name="Id" type="ent:ID" />
        <element name="sObject" type="ens:Opportunity" />
    </sequence>
</complexType>

```

Each object element (in our example, opportunities) contains the subset of the fields that you selected when you [created the outbound message](#). Each message Notification also has the object ID, and this needs to be used to track redelivery attempts of notifications you've already processed.

notificationsResponse

This element is the schema for sending an acknowledgement (ack) response to Salesforce.

```

<element name="notificationsResponse">
    <complexType>
        <sequence>
            <element name="Ack" type="xsd:boolean" />
        </sequence>
    </complexType>
</element> //This section is the last in the types definition section.

```

You acknowledge all notifications in the message if there is more than one.

Building a Listener

Once you have defined an outbound message and configured an outbound messaging endpoint, download the WSDL and create a listener:

1. Right-click **Click for WSDL** and select Save As to save the WSDL to a local directory with an appropriate file name. For example, for an outbound message that deals with leads, you could name the WSDL file `leads.wsdl`.
2. Unlike the enterprise or partner WSDLs, which describe the messages the client sends to Salesforce, this WSDL defines the messages that Salesforce will send to your client application.
3. Most Web services tools will generate stub listeners for you, in much the same way as they generate a client stub for the enterprise or partner WSDL. Look for a server side stub option.

For example, for .Net 2.0:

- a. Run `wsdl.exe /serverInterface leads.wsdl` with .Net 2.0. This generates `NotificationServiceInterfaces.cs`, which defines the notification interface.
- b. Create a class that implements `NotificationServiceInterfaces.cs`.
- c. You implement your listener by writing a class that implements this interface. There are a number of ways to do this. One simple way is to compile the interface to a DLL first (DLLs need to be in the `bin` directory in ASP.NET).

```
mkdir bin  
csc /t:library /out:bin\nsi.dll NotificationServiceInterfaces.cs
```

Now write an ASMX based Web service that implements this interface. For example, in `MyNotificationListener.asmx`:

```
<%@WebService class="MyNotificationListener" language="C#" %>  
class MyNotificationListener : INotificationBinding  
{  
    public notificationsResponse notifications(notifications n)  
    {  
        notificationsResponse r = new notificationsResponse();  
        r.Ack = true;  
        return r;  
    }  
}
```

This example is a simple implementation, actual implementations will be more complex.

- d. Deploy the service by creating a new virtual directory in IIS for the directory that contains the `MyNotificationListener.asmx`.
- e. You can now test that the service is deployed by viewing the service page with a browser. For example, if you create a virtual directory `salesforce`, you'd go to `http://localhost/salesforce/MyNotificationListener.asmx`.

The process for other Web service tools is similar, please consult the documentation for your Web service tool.

Your listener must meet these requirements:

- Must be reachable from the public Internet.
- For security reasons, Salesforce restricts the outbound ports you may specify to one of the following:
 - 80: This port only accepts HTTP connections.
 - 443: This port only accepts HTTPS connections.

- 1024–66535 (inclusive): These ports accept HTTP or HTTPS connections.
- To be valid, the common name (CN) of the certificate must match the domain name for your endpoint's server, and the certificate must be issued by a Certificate Authority trusted by Java 2 Platform, Standard Edition (J2SE) 5.0 (JDK 1.5).
- If your certificate expires, message delivery will fail.



Warning: To avoid an infinite loop of outbound messages that trigger changes that trigger more outbound messages, the user who updates the objects should **not** have the “Send Outbound Messages” permission.

CHAPTER 15 Data Loading and Integration

In this chapter ...

- [Choosing the Right API](#)
- [Client Application Design](#)
- [Salesforce Settings](#)
- [Best Practices with Any Data Loader](#)
- [Integration and Single Sign-On](#)

If you need to load large volumes of data (hundreds of thousands to millions of records), there are a number of factors you must consider. Use the topics in this section to become familiar with issues of choosing an API, client application design, organization configuration, and data loader best practices.

Choosing the Right API

The first decision that you must make is which API to use for your data loading process.

SOAP API

Use SOAP API to create, retrieve, update or delete records, such as accounts, leads, and custom objects. With more than 20 different calls, SOAP API also allows you to maintain passwords, perform searches, and much more. Use SOAP API in any language that supports Web services.

REST API

REST API provides a powerful, convenient, and simple REST-based web services interface for interacting with Salesforce. Its advantages include ease of integration and development, and it's an excellent choice of technology for use with mobile applications and web projects. However, if you have many records to process, consider using Bulk API, which is based on REST principles and optimized for large sets of data.

Bulk API

Bulk API is based on REST principles and is optimized for loading or deleting large sets of data. You can use it to query, insert, update, upsert, or delete many records asynchronously by submitting batches. Salesforce processes batches in the background.

SOAP API, in contrast, is optimized for real-time client applications that update a few records at a time. You can use SOAP API for processing many records, but when the data sets contain hundreds of thousands of records, SOAP API is less practical. Bulk API is designed to make it simple to process data from a few thousand to millions of records.

Client Application Design

Although the Bulk API is the best choice for loading large numbers of records, you can also use the SOAP-based API. There are many ways you can design your application to improve the speed of data loads:

- **Use persistent connections.** Opening a socket takes time, mostly when opening a socket stems from the SSL/TLS negotiation. Without SSL or TLS, the API request would not be secure. Included in the HTTP 1.1 specification is support for reusing sockets among requests (persistent connections) instead of having to re-open a socket per request as in HTTP 1.0. Whether your client supports persistent connections depends on the SOAP stack you are using. By default, .NET uses persistent connections. If you change the configuration to use the Apache http-commons libraries, your client will be compliant with the HTTP 1.1 specification and use persistent connections.

For information about HTTP 1.1, see [HTTP Persistent Connections](http://www.w3.org/Protocols/rfc2616/rfc2616-sec8.html#sec8.1) and <http://www.w3.org/Protocols/rfc2616/rfc2616-sec8.html#sec8.1>.

- **Minimize the number of requests.** There is some processing associated with each request, so to save time your client should batch as many records per request as possible. Set `batchSize` to the limit of 2,000. If that is not the most efficient batch size, the API will change it. For more information about setting batch sizes, see [QueryOptions](#).
- **Minimize the size of the requests.** Your client application should send as many records per request as possible, but it should also send as small a request as possible to reduce network transmission time. To minimize the request size, use compression on both the request and the response. Gzip is the most popular type of compression, and there are multiple posts on the community boards at the [Force.com Developer Boards](#) that describe how to implement compression with different SOAP stacks. The full Gzip analysis and discussion is available at Simon Fell's blog: <http://www.pocketsoap.com/weblog/2005/12/1583.html>.

- **Do Not Design a Multi-Threaded Client Application.** Multi-threading is not allowed for a single client application using the SOAP-based API.

Salesforce Settings

Most processing takes place in the database. Setting these parameters correctly will help the database process as quickly as possible:

- **Enable or Disable the Most Recently Used (MRU) functionality.** Records marked as most recently used (MRU) are listed in the “Recent Items” section of the sidebar in the Salesforce user interface. Check that you are not enabling it for calls where it is not needed.

In API version 7.0 and above, MRU functionality is disabled by default. To enable the MRU functionality, create this header and set the `updateMru` to `true`. The following sample shows how to use MRU functionality:

```
public void mruHeaderSample() {
    connection.setMruHeader(true);
    Account account = new Account();
    account.setName("This will be in the MRU");
    try {
        SaveResult[] sr = connection.create(new SObject[]{account});
        System.out.println("ID of account added to MRU: " +
            sr[0].getId());
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

- **Log in as a user with the “Modify All Data” permission to avoid sharing rules.** If the client application logs in as a user who has access to data via a sharing rule, then the API must issue an extra query to check access. To avoid this, log in as a user with the “Modify All Data” permission. In general, fewer sharing rules quickens load speeds, as there are fewer operations that have to be performed when setting properties such as ownership.

Alternatively, you can set organization-wide defaults for some objects as public read/write for the duration of the load. For more information, see “Set Your Organization-Wide Sharing Defaults” in the Salesforce online help.

- **Avoid workflow or assignment rules.** Anything that causes a post-operation action slows down the load. You can temporarily disable automatic rules if the loaded objects are exempt from them.
- **Avoid triggering cascading updates.** For example, if you update the owner of an account, the contacts and opportunities associated with that account may also require updates. Instead of updating a single object, the client application must access multiple objects, which slows down the load.

The Force.com Data Loader is a good reference for data loading. It disables the MRU, uses HTTP/1.1 persistent connections, and applies GZIP compression on the request and response. If you are performing a data load, or are looking at a place to start when writing your own Java integration, the Force.com Data Loader can serve as a fast and reliable solution. For more information about the Force.com Data Loader, see: Data Loader in the Salesforce online help.

Best Practices with Any Data Loader

While this section presents a best practice process using the Force.com Data Loader, the general principles apply to any client data loader:

1. Identify which data you will migrate.

You may not want or need to migrate a whole set of data—choose which objects you wish to migrate. For example, you may want to migrate only the contact information from each account, or only migrate account information from a particular division.

2. Create templates for the data.

Create one template for each object, for example in an Excel worksheet.

Identify the required fields for each object. In addition to the required fields for each standard object, there may be additional required fields such as those needed to follow business rules, or legacy ID fields. Use this guide or see the page layout definitions in the Salesforce user interface to find out which fields are required on standard objects.

You may wish to highlight the required fields in red for easier review of the data after you populate the templates.

You should also identify any ordering dependencies. Objects may have mandatory relationships, for example all accounts have an owner, and all opportunities are associated with an account. The dependencies in these relationships dictate the order of data migration. For Salesforce data, for example, you should load users first, then accounts, then opportunities.

To identify dependencies, review the related lists and lookup fields in the page layout of the given object, and IDs (foreign keys) in the database.

3. Populate the templates.

Clean your data before populating the template, and review the data in the templates.

4. Migrate the data.

Create custom fields to store legacy ID information. Optionally, give the custom field the `External_ID` attribute so it will be indexed. This will help maintain relationships, and help you build custom reports for validation.

Load one record, check the results, then load all records.

5. Validate the data.

Use all of these techniques to validate your migration:

- Create custom reports that validate record counts and provide an overall snapshot of migration.
- Spot check the data.
- Review exception reports to see what data was not migrated.

6. Re-migrate or update data as needed.

Integration and Single Sign-On



Warning: To avoid getting into an unrecoverable state, do not enable single sign-on for your system administrator account. If you do, and then perform a single sign-on integration that fails, you may not be able to log in again to recover.

CHAPTER 16 Data Replication

In this chapter ...

- [API Calls for Data Replication](#)
- [Scope of Data Replication](#)
- [Data Replication Steps](#)
- [Object-Specific Requirements for Data Replication](#)
- [Polling for Changes](#)
- [Checking for Structural Changes in the Object](#)

The API supports data replication, which allows you to store and maintain a local, separate copy of your organization's pertinent Salesforce data for specialized uses, such as data warehousing, data mining, custom reporting, analytics, and integration with other applications. Data replication provides you with local control and the ability to run large or ad hoc analytical queries across the entire data set without transmitting all that data across the network.

Use the topics in this section to better understand the best practices for data replication.

API Calls for Data Replication

The API supports data replication with the following API calls:

API Call	Description
<code>getUpdated()</code>	Retrieves the list of objects that have been updated (added or changed) during the specified timespan for the specified object.
<code>getDeleted()</code>	Retrieves the list of objects that have been deleted during the specified timespan for the specified object.

Client applications can invoke these API calls to determine which objects in your organization's data have been updated or deleted during a given time period. These API calls return a set of IDs for objects that have been updated (added or changed) or deleted, as well as the timestamp (Coordinated Universal Time (UTC)—not local—timezone) indicating when they were last updated or deleted. It is the responsibility of the client application to process these results and to incorporate the required changes into the local copy of the data.

Scope of Data Replication

This feature provides a mechanism that targets data replication (one-way copying of data). It does not provide data synchronization (two-way copying of data) or data mirroring capabilities.

Data Replication Steps

The following is a typical data replication procedure for an object:

1. Optionally, determine whether the structure of the object has changed since the last replication request, as described in [Checking for Structural Changes in the Object](#).
2. Call `getUpdated()`, passing in the object and timespan for which to retrieve data.
Note that `getUpdated()` retrieves the IDs for data to which the logged in user has access. Data that is outside of the user's sharing model is not returned. The API returns the ID of every changed object that is visible to you, regardless of what change occurred in the object. For information on IDs, see [ID Field Type](#).
3. Pass in all IDs in an array. For each ID element in the array, call `retrieve()` to obtain the latest information you want from the associated object. You must then take the appropriate action on the local data, such as inserting new rows or updating existing ones with the latest information.
4. Call `getDeleted()`, passing in the object and timespan for which to retrieve data. Like `getUpdated()`, `getDeleted()` retrieves the IDs for data to which the logged-in user has access. Data that is outside of the user's sharing model is not returned. The API returns the ID of every changed object that is visible to you, regardless of what change occurred in the object. For information on IDs, see [ID Field Type](#).
5. Iterate through the returned array of IDs. Your client application must then take the appropriate action on the local data to remove (or flag as deleted) the deleted objects. If your client application cannot match rows in the local data using the retrieved object ID, then the local data rows either were deleted or were never created, in which case there is nothing to do.
6. Optionally, save the request time spans for future reference. You can do this with the `getDeleted().latestDateCovered` value or the `getUpdated().latestDateCovered` value.

Object-Specific Requirements for Data Replication

The API objects have the following requirements for data replication:

- The `getUpdated()` and `getDeleted()` calls filter the results so that the client application receives IDs for only those created or updated objects to which the logged-in user has access. For information on IDs, see [ID Field Type](#).
- Your client application can replicate any objects to which it has sufficient permissions. For example, to replicate all data for your organization, your client application must be logged in with the “View All Data” permission. For more information, see [Factors that Affect Data Access](#).
- The logged-in user must have read access to the object. For more information, see “Set Your Organization-Wide Sharing Defaults” in the Salesforce online help.
- The object must be configured to be replicableable (`replicableable` is `true`). To determine whether a given object can be replicated, your application can invoke the `describeSObject()` call on the object and inspect the `replicableable` property in the `describeSObjectResult`.

Polling for Changes

Client applications typically poll for changed data periodically. Polling involves the following considerations:

- The polling frequency depends on business requirements for how quickly changes in your organization’s Salesforce data need to be reflected in the local copy. Some client applications might poll once a day to retrieve changes, while other client applications might poll every five minutes to achieve closer accuracy.
- Deleted records are written to a delete log, which `getDeleted()` accesses. A background process that runs every two hours purges records that have been in an organization’s delete log for more than two hours if the number of records is above a certain limit. Starting with the oldest records, the process purges delete log entries until the delete log is back below the limit. This is done to protect Salesforce from performance issues related to massive delete logs. The limit is calculated using this formula:

```
5000 * number of licenses in the organization
```

For example, an organization with 1,000 licenses could have up to 5,000,000 (five million) records in the delete log before any purging took place. If purging has been performed before your `getDeleted()` call is executed, an `INVALID_REPLICATION_DATE` error is returned. If you get this exception, you should do a full pull of the table.

- The API truncates the seconds portion of `dateTime` values. For example, if a client application submits a timespan between 12:30:15 and 12:35:15 (Coordinated Universal Time (UTC) time), then the API retrieves information about items that have changed between 12:30:00 and 12:35:00 (UTC), inclusive.



Note: Development tools differ in the way that they handle time data. Some development tools report the local time, while others report only the Coordinated Universal Time (UTC) time. To determine how your development tool handles time values, refer to its documentation.

- We recommend polling no more frequently than every five minutes. There are built in controls to prevent errant applications from invoking the data replication API calls too frequently.
- Client applications should save the timespan used in previous data replication API calls so that the application knows the last time period for which data replication was successfully completed.
- To ensure data integrity on the local copy of the data, a client application needs to capture all of the relevant changes during polling—even if it requires processing data redundantly to ensure that there are no gaps. Your client application can contain business logic to skip processing objects that have already been integrated into your local data.

- Gaps can also occur if the client application somehow fails to poll the data as expected (for example, due to a hardware crash or network connection failure). Your client application can contain business logic that determines the last successful replication and polls for the next consecutive timespan.
- If for any reason the local data is compromised, your client application might also provide business logic for rebuilding the local data from scratch.



Note: You can now use [Outbound Messaging](#) to trigger actions instead of polling for them.

Checking for Structural Changes in the Object

In the API, data replication only reflects changes made to object records. It does not determine whether changes have been made to the structure of objects (for example, fields added to—or removed from—a custom object). It is the responsibility of the client application to check whether the structure of a given object has changed since the last update. Before replicating data, client applications can call `describeSObjects()` on the object, and then compare the data returned in the `DescribeSObjectResult` with the data returned and saved from previous `describeSObjects()` invocations.

CHAPTER 17 Feature-Specific Considerations

In this chapter ...

- [Archived Activities](#)
- [Person Account Record Types](#)
- [Opportunity Forecast Override Business Rules](#)
- [External Objects](#)
- [Call Centers and the API](#)
- [Implementing Salesforce Integrations on Force.com](#)
- [Articles](#)
- [Data Categories](#)

Some Salesforce features require special consideration when accessed via the API. Use the topics in this section to learn about the special considerations for activities, person accounts, forecast override business rules, the Call Center, and creating your own apps.

Archived Activities

Salesforce archives activities (tasks and events) that are over a year old.

You can use `queryAll()` to query on all `Task` and `Event` records, archived or not. You can also filter on the `isArchived` field to find only the archived objects. You cannot use `query()` as it automatically filters out all records where `isArchived` is set to `true`. You can update or delete archived records, though you cannot update the `isArchived` field. If you use the API to insert activities that meet the criteria listed below, the activities will be archived during the next run of the archival background process.

Older `Events` and `Tasks` are archived according to the criteria listed below. In the Salesforce user interface, users can view archived activities only in the **Printable View** or by clicking **View All** on the Activity History related list or by doing an advanced search. However, in the API, archived activities can only be queried via `queryAll()`.

Activity archive criteria:

- `Events` with an `ActivityDateTime` or `ActivityDate` value greater than or equal to 365 days old
- `Tasks` with an `IsClosed` value of `true` and an `ActivityDate` value greater than or equal to 365 days old
- `Tasks` with an `IsClosed` value of `true`, a blank `ActivityDate` field, and a create date greater than or equal to 365 days ago

Person Account Record Types

Beginning with API version 8.0, a new family of record types on `Account` objects is available: “person account” record types. The person account record types enable specialized business-to-consumer functionality for those who sell to or do business with individuals. For example, a doctor, hairdresser, or real estate agent whose clients are individuals. For more information about person accounts, see “What Is a Person Account?” and “Person Account Behaviors” in the Salesforce online help.

Record types are person account record types if the `Account` field `IsPersonAccount` is set to `true`. Salesforce provides one default person account record type, `PersonAccount`, but an administrator can create additional person account record types. Conversely, record types with the `Account` field `IsPersonAccount` set to `false` are “business account” record types, which are traditional business-to-business (B2B) Salesforce accounts.

When a person account is created (or an existing business account is changed to a person account), a corresponding contact record is also created. This contact record is referred to as a “person contact.” The person contact enables the person account to function simultaneously as both an account and a contact. This is the one and only contact record that can be associated directly with the person account. Also, the ID of the corresponding person contact record is stored in the `PersonContactId` field on the person account.

Review this list of facts about person account record types before working with them.

- You may need to contact your account representative to enable the person account feature.
- You can use a query similar to the following example to find all records with a person account record type:

```
SELECT Name, SobjectType, IsPersonType  
FROM RecordType  
WHERE SobjectType='Account' AND IsPersonType=True
```

- If you issue a `query()` call against an account, the results return the root object type in the `SobjectType` field. This means that the value returned will always be `Account`.
- A person contact can be modified, but cannot be created or deleted. Since these kinds of contacts do not have their own record detail page, clients should redirect users to the corresponding person account (`Account`) page. SOSL results will not include any of the contact fields enabled when `IsPersonAccount` is set to `true`. The contact `ReportsToId` field is not visible.
- If you delete the account, the contact is also deleted. You cannot directly delete the contact; you must delete the account.

- You can change the record type of an account across record type families (typically performed when migrating business accounts to person accounts, but the reverse operation is also supported). When you change the record type from a business account to a person account, the person contact is created. When you change the record type from a person account to a business account, the person fields are set to null, and the person contact becomes a regular contact with the same parent account it had before the change.



Note: You cannot change record types across record type families in the Salesforce user interface.

- If you change the record type of a business account to a person account using either `update()` or `upsert()`, you cannot make any other changes to fields in that account in the same call; if attempted, the fault `INVALID_FIELD_FOR_INSERT_UPDATE` will result. However, you can change record type values from one person account record type to another, or from one business account record type to another, in the same call with other changes.
- When converting a business account to a person account, there must be a one-to-one relationship between each business account record and its corresponding contact record. Furthermore, fields common to both records such as `Owner` and `Currency` must have identical values.
- Workflow and validation formulas do not fire during a change in record types from or to person accounts.
- When you change a business account to a person account, valid records will be changed and invalid records will show an error in the results array.
- When you change a person account to a business account, no validation is performed.
- `describeLayout()` for version 7.0 and below will return the default business account record type as the default record type even if the tab default is a person account record type. In version 8.0 and after, it will always be the tab default.
- `describeLayout()` for version 7.0 and below will not return any person account record types.
- `describeSObject()` for version 7.0 and below will show `Account` objects as not createable if the profile does not have access to any business record types.
- After conversion, the new person accounts will have unique one-to-one relationships with the contact records that formed them. As is true for all person accounts, no other contacts can be associated to a person account.
- After conversion, any existing account field history information remains on the person accounts. Any existing contact field history information is retained on the contact, but is not added to the person accounts field history.

For more information about person accounts, see the Salesforce online help.

Opportunity Forecast Override Business Rules

Customizable forecasting is the preferred way to track revenue data in Salesforce. If you have customizable forecasting enabled for your organization, users with the “Override Forecast” permission can override forecast amounts for themselves and their direct reports. For more information, see “Overriding Customizable Forecasts” in the Salesforce online help.

You can use the following objects to retrieve opportunity forecast override information.

- `OpportunityOverride`
- `LineitemOverride`

These read-only objects keep an up-to-date record of the current values related to forecasts, whether those values are inherited directly from the opportunity or reflect an opportunity forecast override. Inherited values are stored in these objects, separate from the opportunity object, to improve performance.

The forecast is computed differently depending on whether or not there is an override record.

- If an override record exists (whether or not a particular value is overridden), the following values are used to compute the forecast for the `OwnerId` specified in the record.

- Amount
 - Quantity
 - Forecast Period
 - Forecast Category
 - Unit Price on LineitemOverride
 - OpportunityOverride or LineitemOverride. These objects represent the user's view of the [Opportunity](#) or [OpportunityLineItem](#) with regard to forecasting, and may reflect overridden values from an opportunity owner or Forecast Manager below the specified user in the Forecast Hierarchy.
- If an override record does not exist, then the values stored in the Opportunity or OpportunityLineItem record are used, and no override information appears to the user.

In the Forecast Hierarchy, one user at each non-leaf level is designated as the Forecast Manager. The Forecast Manager sees the Opportunity Forecast Related List on the Opportunity Detail page when he or she views an opportunity owned by a subordinate user (if that user has the Allow Forecasting permission). A Forecast Manager's forecasts include contributions from those same subordinates. Other users in the same role as a Forecast Manager only see their own opportunities in their forecasts.

 **Note:** Keep in mind that this section discusses both forecast managers and opportunity owners. When we describe traversals from a user and travelling along the forecast hierarchy, it is always in order, unless otherwise specified. For more information about forecast managers and the forecast hierarchy, see the Salesforce online help.

If your organization has territory management enabled, the territory hierarchy drives forecast data. See Territory Management in the Salesforce online help.

OpportunityOverride Lifecycle

[OpportunityOverride](#) records are created, updated, or deleted only under certain conditions, and only for relevant users.

 **Note:** Because [LineitemOverride](#) records always have a parent record in the OpportunityOverride object with the same [OpportunityId](#) and [OwnerId](#), this section explains the lifecycle of the OpportunityOverride records first, then explains any additional behaviors relevant only to LineitemOverride records.

Creating Opportunity Overrides

Forecast managers can override a forecast-related value (such as [Amount](#), [Quantity](#), [Forecast Period](#), and [ForecastCategory](#)) by editing the Opportunity Forecasts related list on an opportunity owned by a subordinate user. Also, the opportunity owner can change [Opportunity](#) forecast-related values (such as [ForecastCategory](#), [StageName](#), [Amount](#), and [CloseDate](#)) on an Opportunity that he or she owns. [ForecastCategory](#) has additional rules governing it; see [Special Case: Forecast Category](#) for more information.

When a user sets the first override for a particular Opportunity record, [OpportunityOverride](#) records are created and appropriate values are written for the opportunity owner and every forecast manager above the Opportunity owner in the forecast hierarchy. The values stored in each user's record depend on their location in the hierarchy—users below the user who made an override will have the original values. Other users will have the overrides, plus the Opportunity values for anything not overridden.

When subsequent override values are set, the new values are written to the OpportunityOverride records of the user who set the override and other forecast managers above in the forecast hierarchy, until an existing override value is reached.

Quantity can only be overridden in the Salesforce user interface if Quantity Forecasting is enabled on the Forecast Settings page, and [Amount](#) can only be overridden if Revenue Forecasting is enabled.

 **Note:** Any changes that an opportunity owner can make on the Opportunity Forecast edit page are applied to the Opportunity record as well as the relevant OpportunityOverride records.

Updating Opportunity Values and Opportunity Override Values

Whenever an opportunity forecast-related value is updated on the [Opportunity](#) record itself, the corresponding records in the [OpportunityOverride](#) for the Opportunity owner and forecast managers above the owner in the forecast hierarchy are updated, until the first override value is encountered. The Opportunity owner cannot override anything except `ForecastCategory`, but he or she can edit the `CloseDate` and `StageName` on the Opportunity Forecast edit page, in addition to `ForecastCategory`. All three fields correspond to the Opportunity record, and the result is the same whether they make the changes on the Opportunity Forecast edit page or the Opportunity edit page.

Changes to the following Opportunity fields also affect records in the OpportunityOverride.

- `CloseDate` affects the forecast `PeriodId` if the new `CloseDate` falls into a different Forecast Period than the old `CloseDate`. In this case, the `OpportunityOverride PeriodId` is updated on the opportunity override record of the opportunity owner, and all forecast managers above the owner in the forecast hierarchy, until the first overridden `PeriodId` is encountered.
- [CurrencyType IsoCode](#) changes always cause an update to the opportunity owner's `OpportunityOverride`. No other object is updated unless `Amount` changes, which is often the case.
- If territory management is enabled, territory changes on the Opportunity affect `OpportunityOverride` records. An Opportunity owner's `OpportunityOverrides` are updated with the new territory, and `OpportunityOverride` records are inserted or updated for forecast managers above the Opportunity owner in the forecast hierarchy. `OpportunityOverride` records are also deleted for any forecast manager whose territory (stored on the override record) is no longer above the Opportunity owner in the new opportunity territory. That is, even if the Opportunity owner remains a subordinate of a forecast manager before and after the update, if they roll up into a different territory, that forecast manager loses the `OpportunityOverrides` for the old territory and new records are created for the new territory.
- When an Opportunity is set to `Closed Won` (the `StageName` value is `Closed Won`), the `AmountInherited`, `QuantityInherited`, and `PeriodInherited` fields in `OpportunityOverride` are set to `true` and the corresponding values are updated to match the Opportunity, if they are different. The `OpportunityOverride ForecastCategoryInherited` flag is also set to `true` and the Opportunity `ForecastCategory` is set to `Closed`, unless it has been overridden to `Omitted`. `Omitted` is the only valid `ForecastCategory` override for a `Closed Won` opportunity.
- When an Opportunity is set to `Closed Lost`, the `Amount` and `PeriodInherited` flags are set to `true`, and the corresponding values are updated to match the Opportunity, if they are different, and the `ForecastCategoryInherited` flag is set to `true` and the `ForecastCategory` is set to `Omitted` if the current value is anything other than `Omitted`.

Deleting Opportunity Overrides

[OpportunityOverride](#) records are not deleted unless one of the following triggering events occurs.

- An [Opportunity](#) is deleted.
- A change is made in the forecast hierarchy, such that a particular `OpportunityOverride` owner is no longer above the Opportunity owner or is no longer the forecast manager in the relevant role or territory.
- An Opportunity is transferred to a new owner or territory, individually or as part of an account transfer.
- The "Allow Forecasting" permission is removed from a user.

When an Opportunity is transferred to a new owner, an `OpportunityOverride` record for the new owner is added as needed, and `OpportunityOverride` records are deleted for all users that are not forecast managers above the new owner in the forecast hierarchy, possibly including the previous Opportunity owner. Also, `OpportunityOverride` records are updated or inserted as necessary for all forecast managers above the new owner in the forecast hierarchy. Restrictions on Opportunity owners are strictly enforced during a transfer. If the new owner was in the forecast hierarchy of the previous owner, and had made overrides that are no longer valid as the opportunity owner, such as an `Amount` override, then the corresponding `inherited` flag is set to `true` and the value is refreshed from the opportunity.

Special Case: Forecast Category

An additional rule applies to `ForecastCategory` values: If you set the `ForecastCategory` in a new opportunity that is not the default forecast category for the stage selected, or update an existing `ForecastCategory` on an opportunity, then `OpportunityOverride` records are created for relevant users as described above. Since only `Opportunity` owners can update the `ForecastCategory` on an opportunity record directly, this scenario is treated as a `ForecastCategory` override by the opportunity owner (the `ForecastCategoryInherited` flag on that user's `OpportunityOverride` record is `false`).

LineitemOverride Object Lifecycle

Additional rules apply to line item overrides, as explained in the rest of this section.

Creating Line Item Overrides

The full set of `LineitemOverride` records for each `OpportunityOverride` record is always created to mirror the line item records that exist for the `Opportunity` record. Whether a user makes an override at the opportunity level or line item level, or even just edits Forecast Category on the Opportunity Edit (only opportunity owners can do this), the end result is the same in terms of the Opportunity and `LineitemOverride` records that are created.

Updating Line Item Overrides

All information from [Updating Opportunity Values and Opportunity Override Values](#) applies to updating `LineitemOverride` records, with the following changes.

- Forecast Category overrides at the opportunity level are cascaded down into child `LineitemOverride` records, if the Forecast Category has not been overridden for a particular line item by that user or a subordinate user. That is, when a user establishes an opportunity-level Forecast Category override, any line item-level Forecast Category overrides by a subordinate forecast manager will take precedence when updating line item Forecast Category values for this user and any superiors. The converse is not true. Forecast Category on an `OpportunityOverride` record is never updated in response to a Forecast Category override on a `LineitemOverride` record by any user. When a user sets a Forecast Category override at the line item level, the override value is written to the `LineitemOverride` records of the user who set the override and other forecast managers above that user in the forecast hierarchy, until an existing override value is reached.
- Opportunity line item values for `Unit Price`, `Total Price` and `Quantity`, if changed, are saved on the `LineitemOverride` records for the opportunity owner and above in the forecast hierarchy, until the first overriden value is encountered for the field. `Quantity` can only be overridden in the Salesforce user interface if Quantity Forecasting is enabled on the Forecast Settings page. `Unit Price` and `Total Price` can only be overridden if Revenue Forecasting is enabled.

When `Quantity` or `Unit Price` are overridden, the `Total Price` is computed and written to the relevant `LineitemOverride` record. When `Total Price` is overridden, but not `Unit Price`, the `Unit Price` is computed and written to the relevant `LineitemOverride` records. If both are overridden, no computation occurs. The computation applies to the `LineitemOverride` record that contains the override value and related `LineitemOverride` records that inherit the overridden value. That is, for a given `LineitemOverride` record, we compute `Total Price` or `Unit Price` according to the above rules, if an override is made on the current record or if the record inherited an override made by a subordinate forecast manager.

If you selected the value **Schedule Date** in Forecasts Settings, you can't override the following values.

- If forecasting by Revenue, or Revenue and Quantity, and the `OpportunityLineItem` record has a Revenue schedule, you cannot override `Unit Price` and `Total Price`.
- If forecasting by Quantity, or Revenue and Quantity, and the `OpportunityLineItem` records have Quantity schedules, you cannot override `Quantity`.

Deleting Line Item Overrides

Whenever a line item is deleted, `LineitemOverride` records relating to that line item are deleted for all users, similar to opportunities. Also, opportunity transfers and forecast hierarchy changes affect line item overrides the same way they affect opportunity overrides.

That is, if an OpportunityOverride record is deleted, all the child LineitemOverride records, matching on `OpportunityId` and `OwnerId`, are also deleted.

External Objects

Special behaviors and limitations apply to `queryAll()` and `queryMore()` calls on external data.

`queryAll()`

Because Salesforce doesn't track changes to external data, `queryAll()` behaves the same as `query()` for external objects.

`queryMore()`

It's common for Salesforce Connect queries of external data to have a large result set that's broken into smaller batches or pages. When querying external objects, Salesforce Connect accesses the external data in real time via Web service callouts. Each `queryMore()` call results in a Web service callout. The batch boundaries and page sizes depend on your adapter and how you set up the external data source.

We recommend the following:

- When possible, avoid paging by filtering your queries of external objects to return fewer rows than the batch size, which by default is 500 rows. Remember, obtaining each batch requires a `queryMore()` call, which results in a Web service callout.
- If the external data frequently changes, avoid using `queryMore()` calls. If the external data is modified between `queryMore()` calls, you can get an unexpected `QueryResult`.

If the primary or "driving" object for a `SELECT` statement is an external object, `queryMore()` supports only that primary object and doesn't support subqueries.

By default, the OData 2.0 and 4.0 adapters for Salesforce Connect use client-driven paging. With client-driven paging, OData adapters convert each `queryMore()` call into an OData query that uses the `$skip` and `$top` system query options to specify the batch boundary and page size. These options are similar to using `LIMIT` and `OFFSET` clauses to page through a result set.

If you enable server-driven paging on an external data source, Salesforce ignores the requested page sizes, including the default `queryMore()` batch size of 500 rows. The pages returned by the external system determine the batches.

Call Centers and the API

The API provides access to information about computer–telephony integration (CTI) call centers with the `describeSoftphoneLayout()` call. You must have the CTI feature enabled for your organization. Contact your account representative for assistance.

The API supports limited access to call center-related objects, including being able to create call centers, and create or modify additional numbers for the call center.

Topic	Description
CallCenter	Call Center object description, including fields and usage.
AdditionalNumber	Configuration settings that allow you to add an additional number if it cannot easily be categorized as a user, contact, lead, account, or any other object. Examples include phone queues or conference rooms.

In addition, several fields have been added to existing objects to support call centers. The following fields provide configuration settings for operation of a call center.

Object Name	Field Name	Field Type	Field Properties	Description
OpenActivity ActivityHistory Task	CallDisposition	string	Create (Task only) Filter Nillable Update (Task only)	Represents the result of a given call, for example, "we'll call back," or "call unsuccessful." Limit is 255 characters. For the Task object, corresponds to the Salesforce user interface label Call Result . You can also create and update values for this field in Task.
OpenActivity ActivityHistory Task	CallDurationInSeconds	int	Create (Task only) Filter Nillable Update (Task only)	Duration of the call in seconds. For Task, you can also create and update values for this field.
OpenActivity ActivityHistory Task	CallObject	string	Filter Nillable Update (Task only)	Name of a call center. Limit is 255 characters. For Task, you can also create and update values for this field.
OpenActivity ActivityHistory Task	CallType	picklist	Create (Task only) Filter Nillable Restricted picklist Update	The type of call being answered: Inbound, Internal, or Outbound. For Task, you can also create and update values for this field.
User	CallCenterId	reference	Create Filter Nillable Update	The unique identifier for the call center associated with this user.

Object Name	Field Name	Field Type	Field Properties	Description
User	UserPermissionsCallCenterAutoLogin	boolean	Create Update	Indicates whether a user will be automatically logged in to a call center when logging in to the Salesforce application (<code>true</code>) or not (<code>false</code>).

Implementing Salesforce Integrations on Force.com

You can implement your Salesforce integrations or other client applications, on the Force.com platform by creating a Force.com AppExchange app.

1. Create a [WebLink](#) that passes the user session ID and the API server URL to an external site:

```
https://www.your_tool.com/test.jsp?sessionid={!API_Session_ID}&url={!API_Partner_Server_URL_80}
```

Use `https` to ensure your session ID cannot be detected.

2. The page pointed to in the step above takes the session ID and uses it to call back to the API. Use `getUserInfo()` to return the `userID` associated with the session and related information. If needed, you can also use `retrieve` on the User object to retrieve any additional information you need about the user.
3. Maintain a cross-reference between the `UserID` or username and the corresponding user ID in your system, which you can do using a [WebLink](#) that is executed when the user clicks on a tab, or a [WebLink](#) on the page layout.
4. Package and upload this app using the instructions in the Salesforce online help topic "Prepare Your Apps for Distribution."

Accessing Salesforce Data Using the API and OAuth

Salesforce supports OAuth versions 1.0A and 2.0 for SOAP API requests.

For OAuth version 1.0.A, after a third-party has successfully negotiated a session with Salesforce (using an already defined connected app and the OAuth protocol) and has a valid AccessToken, a client application can request to access Salesforce data using the API.

The following contains more detailed steps regarding accessing data for developers who are using a connected app to request Salesforce data.

1. The consumer makes an HTTPS POST request to Salesforce.
 - The URL must have the following format:
`https://login.salesforce.com/services/OAuth/type/api-version.`
`type` must have one of the following values.
 - `u`—Partner WSDL
 - `c`—Enterprise WSDL`api-version` must be a valid API version.
 - The authorization header must have the following parameters.
 - `oauth_consumer_key`
 - `oauth_token`
 - `oauth_signature_method`

- oauth_signature
- oauth_timestamp
- oauth_nonce
- oauth_version (optional, must be "1.0" if included)

2. Salesforce validates the request and sends a valid session ID to the consumer. The response header includes the following.

```
<response>

<metadataServerUrl>https://yourInstance.salesforce.com/services/Soap/m/17.0/00D300000006qjK
</metadataServerUrl>
<sandbox>false</sandbox>
<serverUrl>https://yourInstance.salesforce.com/services/Soap/u/17.0/00D300000006qjK
</serverUrl>
<sessionId>00D30000006qrN!AQoAQJTMzwTa67tGgQck1ng_xgMSuWVBpFwZ1xUq2kLjMYg6Zq
GTS8Ezu_C3w0pdT1DMyHiJgB6fbhhEPxKjGqlYnlROIUs1</sessionId>
</response>
```

After a consumer using OAuth version 2.0 has an access token, the method of using the token depends on the API being used.

- For the REST API, use an HTTP authorization header with the following format `Authorization: Bearer Access-Token`.
- For the SOAP API, the access token is placed in the SessionHeader SOAP authentication header.
- For the identity URL, use either an HTTP authorization header (as with the REST API) or use as an HTTP parameter `oauth_token`.

For more information, see [Authenticating Apps with OAuth](#) in the Salesforce online help.

Partners, who wish to get an OAuth consumer Id for authentication, can contact Salesforce

Articles

Articles capture information about your company's products and services that you want to make available in your knowledge base. Articles in the knowledge base can be classified by using one or more [data categories](#) to make it easy for users to find the articles they need. Administrators can use data categories to control access to articles.

Articles are based on article types, which rely on:

- Article-type layouts to organize the content in sections.
- Article-type templates to render articles.

Every article is managed in a publishing cycle.

Article Type

All articles in Salesforce Knowledge are assigned to an *article type*. An article's type determines the type of content it contains, its appearance, and which users can access it. For example, a simple FAQ article type might have two custom fields, `Question` and `Answer`, where article managers enter data when creating or updating FAQ articles. A more complex article type may require dozens of fields organized into several sections. Using layouts and templates, administrators can structure the article type in the most effective way for its particular content. User access to article types is controlled by permissions. For each article type, an administrator can grant "Create," "Read," "Edit," or "Delete" permissions to users. For example, the article manager might want to allow internal users to read, create, and edit FAQ article types, but let partner users only read FAQs.

Article-Type Layout

An *article-type layout* enables administrators to create sections that organize the fields on an article, as well as choose which fields users can view and edit. One layout is available per article type. Administrators modify the layout from the article-type detail page.

Article-Type Template

An *article-type template* specifies how the sections in the article-type layout are rendered. An article type can have a different template for each of its four channels. For example, if the Customer Portal channel on the FAQ article-type is assigned to the Tab template, the sections in the FAQ's layout appear as tabs when customers view an FAQ article. For the Table of Contents template, the sections defined in the layout appear on a single page (with hyperlinks) when the article is viewed. Salesforce provides two standard article-type templates, Tab and Table of Contents. Custom templates can be created with Visualforce.

Channel

A channel refers to the medium by which an article is available. Salesforce Knowledge offers four channels where you can make articles available.

- Internal App: Salesforce users can access articles in the Articles tab depending on their role visibility.
- Customer: Customers can access articles if the Articles tab is available in a community or Customer Portal. Customer users inherit the role visibility of the manager on the account. In a community, the article is only available to users with Customer Community or Customer Community Plus licenses.
- Partner: Partners can access articles if the Articles tab is available in a community or partner portal. Partner users inherit the role visibility of the manager on the account. In a community, the article is only available to users with Partner Community licenses.
- Public Knowledge Base: Articles can be made available to anonymous users by creating a public knowledge base using the *Sample Public Knowledge Base for Salesforce Knowledge* app from the AppExchange. Creating a public knowledge base requires Sites and Visualforce.

Publishing Cycle

Salesforce Knowledge Articles move through a publishing cycle from their creation to their deletion. The publishing cycle includes three different statuses: `Draft` is the stage when a new article is being created or an existing one is being updated. Articles with the `Online` status are draft articles that have been published and are now available to their different channels. Eventually, when a published article is at the end of its life, it can be moved to the `Archived` status or sent back to `Draft` to be updated in a subsequent version.

Working with Articles in the API

Articles are available through the `KnowledgeArticleVersion` and `KnowledgeArticle` objects in the API. They both represent an article but provide different capabilities.

KnowledgeArticleVersion

Every new draft article in Salesforce Knowledge has a version number. When an article is published and you want to update it, you can create a new `Draft` with a distinct version number. Each version has its own ID. Once the updated version is ready to be published, it replaces the former one and updates the version number. You can access the content of an article version using the `KnowledgeArticleVersion` object and filter on its `Draft` or `Online` status. For example, the following query returns the title of the `Draft` version of all the articles across all article types in United States English:

```
SELECT Title  
FROM KnowledgeArticleVersion  
WHERE PublishStatus='Draft'  
AND language ='en_US'
```

You can change the language to return any other language your knowledge base supports, see "Support a Multilingual Knowledge Base" in the Salesforce online help.

Articles are also auto-assigned an Article Number, which is not a unique identifier to an individual article, but an identifier to a master article and all of its available translations.



Note: Both the master version (the Knowledge article with IsMasterLanguage = 1) and the translations are KnowledgeArticleVersion objects.

KnowledgeArticle

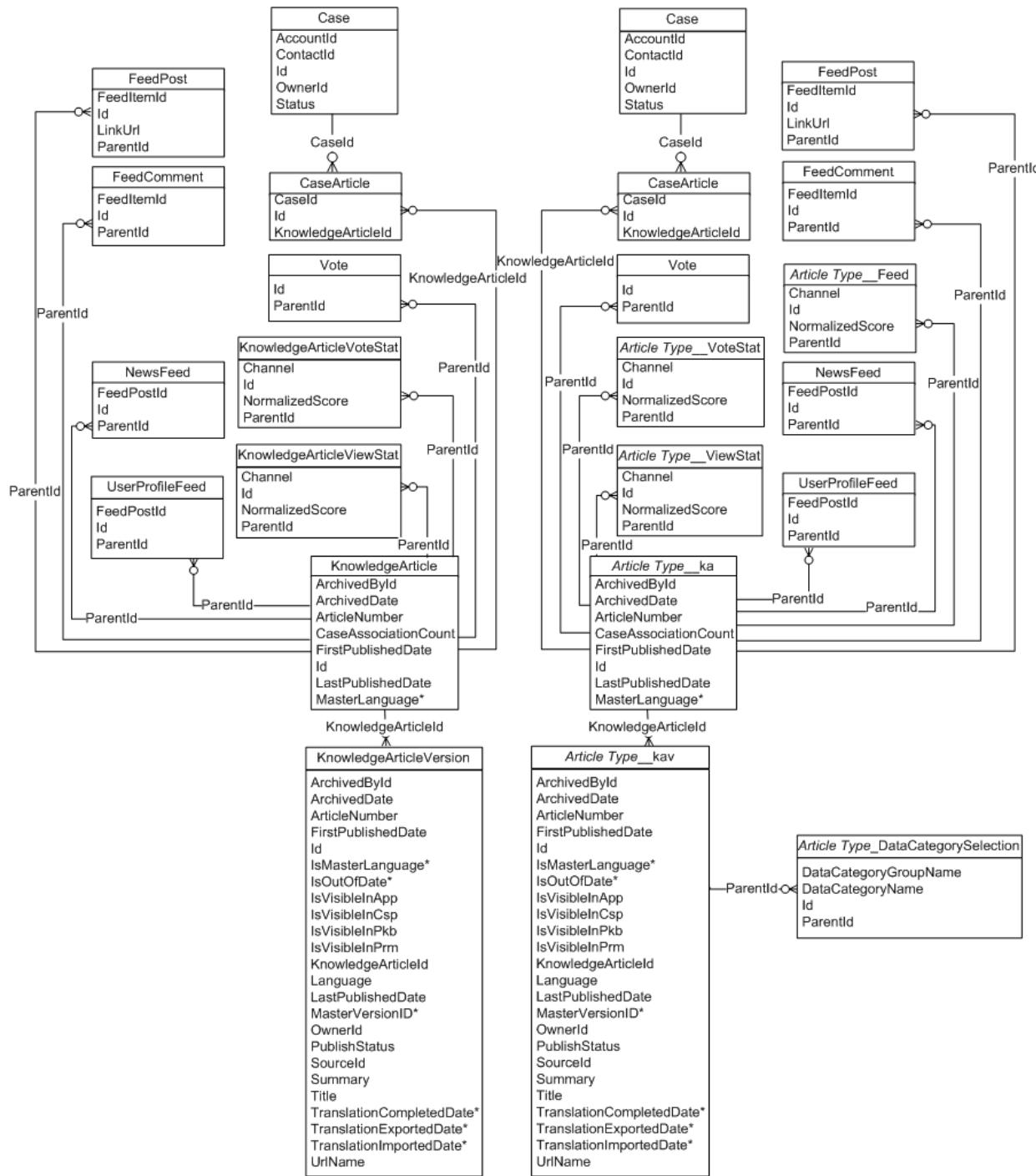
Unlike KnowledgeArticleVersion, the ID of a KnowledgeArticle record is identical irrespective of the article's version (status). Where the KnowledgeArticleVersion object provides API access to an article's custom field values, the KnowledgeArticle object provides API access to an article's metadata fields.

The article record is the parent container of all versions of an article, whatever the publishing status (draft, published, archived) and the language. While KnowledgeArticle and KnowledgeArticleVersion represent any article in the Knowledge Base, <Article Type>_ka and <Article Type>_kav are the concrete representation of respectively KnowledgeArticle ("__ka" suffix) and KnowledgeArticleVersion ("__kav" suffix) for a specific article type. For example, Offer_kav gives access to every Offer articles. KnowledgeArticle and <Article Type>_ka give access to an article independent of its version. KnowledgeArticleVersion and <ArticleType>_kav enables you to retrieve a specific article version and all its standard (KnowledgeArticleVersion) and custom (<ArticleType>_kav) fields. The following query returns the title for all the published offers in United States English:

```
SELECT Title  
FROM Offer_kav  
WHERE PublishStatus='online'  
AND language ='en_US'
```

Salesforce Knowledge Objects

This entity relationship diagram (ERD) illustrates relationships between the Salesforce Knowledge objects.



For field definitions, see its object's page within this guide. Fields with an asterisk (*) are only accessible if your knowledge base supports multiple languages, see "Support a Multilingual Knowledge Base" in the Salesforce online help.

Data Categories

Data categories are organized by category group and let:

- Users classify and find records.
- Administrators control access to records.

Data categories can be used by Salesforce Knowledge (articles) and answers communities (questions).

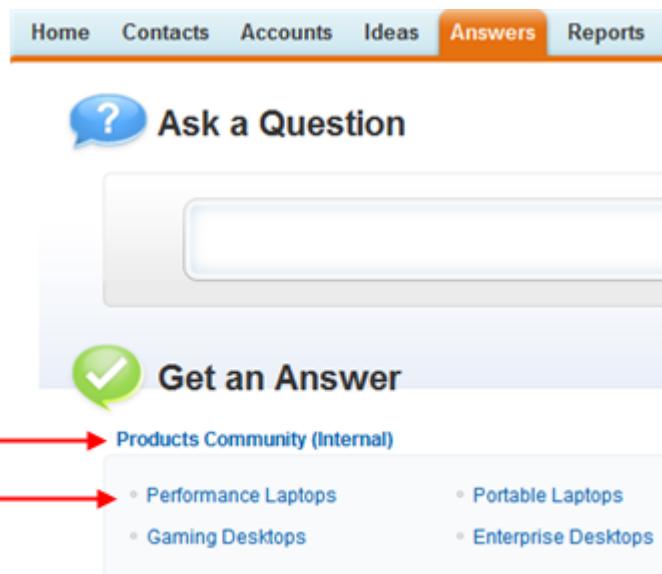
Data Categories and Articles

Salesforce Knowledge uses data categories to classify articles and make them easier to find. For example, to classify articles by sales regions and products, create two category groups: Sales Regions and Products. The Sales Regions category group could consist of a geographical hierarchy, such as All Sales Regions as the top level and North America, Europe, and Asia at the second level. The Products group could have All Products as the top level and Phones, Computers, and Printers at the second.

Data Category Limits	Details
Maximum number of data category groups and active data category groups	5 category groups, with 3 groups active at a time
Maximum number of categories per data category group	100 categories in a data category group
Maximum number of levels in data category group hierarchy	5 levels in a data category group hierarchy
Maximum number of data categories from a data category group assigned to an article	8 data categories from a data category group assigned to an article

Data Categories and Answers

In an answers zone, data categories help organize questions for easy browsing. Each answers zone supports one category group. For example, if you're a computer manufacturer you might create a Products category group that has four sibling categories: Performance Laptops, Portable Laptops, Gaming Desktops, and Enterprise Desktops. On the Answers tab, zone members can assign one of the four categories to each question and then browse these categories for answers to specific questions.



Working with Data Categories in the API

The following table lists API resources for working with data categories.

Name	Type	Description
<code>Article</code>	Object	Gives access to article categorization.
<code>Type__DataCategorySelection</code>	Object	Gives access to question categorization.
<code>WITH DATA CATEGORY filteringExpression</code>	SOQL clause	Filters articles depending on their status in the publishing cycle and their data categories. For more information, see the Salesforce SOQL and SOSL Reference Guide .
<code>WITH DATA CATEGORY DataCategorySpec</code>	SOSL clause	Finds articles based on their categorization. For more information, see the Salesforce SOQL and SOSL Reference Guide .
<code>describeDataCategoryGroups()</code>	Call	Retrieves available category groups for objects specified in the request.
<code>describeDataCategoryGroupStructures()</code>	Call	Retrieves available category groups along with their data category structure for objects specified in the request.
<code>describeDataCategoryGroups</code>	Apex method	Returns a list of the category groups associated with the specified objects. See the Apex Developer Guide .
<code>describeDataCategoryGroupStructures</code>	Apex method	Returns available category groups along with their data category structure for objects specified in the request. See the Apex Developer Guide .

GLOSSARY

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [J](#) | [K](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [Q](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#) | [Y](#) | [Z](#)

A

AJAX Toolkit

A JavaScript wrapper around the API that allows you to execute any API call and access any object you have permission to view from within JavaScript code. For more information, see the [AJAX Toolkit Developer's Guide](#).

Anonymous Block, Apex

Apex code that does not get stored in Salesforce, but that can be compiled and executed through the use of the `ExecuteAnonymousResult()` API call, or the equivalent in the AJAX Toolkit.

Anti-Join

An anti-join is a subquery on another object in a `NOT IN` clause in a SOQL query. You can use anti-joins to create advanced queries. See also Semi-Join.

Apex

Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Force.com platform server in conjunction with calls to the Force.com API. Using syntax that looks like Java and acts like database stored procedures, Apex enables developers to add business logic to most system events, including button clicks, related record updates, and Visualforce pages. Apex code can be initiated by Web service requests and from triggers on objects.

Apex-Managed Sharing

Enables developers to programmatically manipulate sharing to support their application's behavior. Apex-managed sharing is only available for custom objects.

App

Short for "application." A collection of components such as tabs, reports, dashboards, and Visualforce pages that address a specific business need. Salesforce provides standard apps such as Sales and Call Center. You can customize the standard apps to match the way you work. In addition, you can package an app and upload it to the AppExchange along with related components such as custom fields, custom tabs, and custom objects. Then, you can make the app available to other Salesforce users from the AppExchange.

AppExchange

The AppExchange is a sharing interface from Salesforce that allows you to browse and share apps and services for the Force.com platform.

AppExchange Upgrades

Upgrading an app is the process of installing a newer version.

Application Programming Interface (API)

The interface that a computer system, library, or application provides to allow other computer programs to request services from it and exchange data.

B

Boolean Operators

You can use Boolean operators in report filters to specify the logical relationship between two values. For example, the AND operator between two values yields search results that include both values. Likewise, the OR operator between two values yields search results that include either value.

Bulk API

The REST-based Bulk API is optimized for processing large sets of data. It allows you to query, insert, update, upsert, or delete a large number of records asynchronously by submitting a number of batches which are processed in the background by Salesforce. See also SOAP API.

C

Callout, Apex

An Apex callout enables you to tightly integrate your Apex with an external service by making a call to an external Web service or sending a HTTP request from Apex code and then receiving the response.

Child Relationship

A relationship that has been defined on an sObject that references another sObject as the “one” side of a one-to-many relationship. For example, contacts, opportunities, and tasks have child relationships with accounts.

See also sObject.

Class, Apex

A template or blueprint from which Apex objects are created. Classes consist of other classes, user-defined methods, variables, exception types, and static initialization code. In most cases, Apex classes are modeled on their counterparts in Java.

Client App

An app that runs outside the Salesforce user interface and uses only the Force.com API or Bulk API. It typically runs on a desktop or mobile device. These apps treat the platform as a data source, using the development model of whatever tool and platform for which they are designed.

Component, Visualforce

Something that can be added to a Visualforce page with a set of tags, for example, `<apex:detail>`. Visualforce includes a number of standard components, or you can create your own custom components.

Component Reference, Visualforce

A description of the standard and custom Visualforce components that are available in your organization. You can access the component library from the development footer of any Visualforce page or the [Visualforce Developer's Guide](#).

Controller, Visualforce

An Apex class that provides a Visualforce page with the data and business logic it needs to run. Visualforce pages can use the standard controllers that come by default with every standard or custom object, or they can use custom controllers.

Controlling Field

Any standard or custom picklist or checkbox field whose values control the available values in one or more corresponding dependent fields.

Custom App

See App.

Custom Field

A field that can be added in addition to the standard fields to customize Salesforce for your organization’s needs.

Custom Help

Custom text administrators create to provide users with on-screen information specific to a standard field, custom field, or custom object.

Custom Links

Custom links are URLs defined by administrators to integrate your Salesforce data with external websites and back-office systems. Formerly known as Web links.

Custom Object

Custom records that allow you to store information unique to your organization.

Custom S-Control

 **Note:** S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won't be allowed to create them. Existing s-controls will remain unaffected, and can still be edited.

Custom Web content for use in custom links. Custom s-controls can contain any type of content that you can display in a browser, for example a Java applet, an Active-X control, an Excel file, or a custom HTML Web form.

D

Database

An organized collection of information. The underlying architecture of the Force.com platform includes a database where your data is stored.

Database Table

A list of information, presented with rows and columns, about the person, thing, or concept you want to track. See also Object.

Data Loader

A Force.com platform tool used to import and export data from your Salesforce organization.

Data Manipulation Language (DML)

An Apex method or operation that inserts, updates, or deletes records.

Date Literal

A keyword in a SOQL or SOSL query that represents a relative range of time such as `last month` or `next year`.

Decimal Places

Parameter for number, currency, and percent custom fields that indicates the total number of digits you can enter to the right of a decimal point, for example, 4.98 for an entry of 2. Note that the system rounds the decimal numbers you enter, if necessary. For example, if you enter 4.986 in a field with `Decimal Places` of 2, the number rounds to 4.99. Salesforce uses the round half-up rounding algorithm. Half-way values are always rounded up. For example, 1.45 is rounded to 1.5. –1.45 is rounded to –1.5.

Delegated Authentication

A security process where an external authority is used to authenticate Force.com platform users.

Dependent Field

Any custom picklist or multi-select picklist field that displays available values based on the value selected in its corresponding controlling field.

Developer Edition

A free, fully-functional Salesforce organization designed for developers to extend, integrate, and develop with the Force.com platform. Developer Edition accounts are available on developer.salesforce.com.

Glossary

Salesforce Developers

The Salesforce Developers website at developer.salesforce.com provides a full range of resources for platform developers, including sample code, toolkits, an online developer community, and the ability to obtain limited Force.com platform environments.

Document Library

A place to store documents without attaching them to accounts, contacts, opportunities, or other records.

E

Email Alert

Email alerts are actions that send emails, using a specified email template, to specified recipients.

Email Template

A form email that communicates a standard message, such as a welcome letter to new employees or an acknowledgement that a customer service request has been received. Email templates can be personalized with merge fields, and can be written in text, HTML, or custom format.

Enterprise Edition

A Salesforce edition designed for larger, more complex businesses.

Enterprise WSDL

A strongly-typed WSDL for customers who want to build an integration with their Salesforce organization only, or for partners who are using tools like Tibco or webMethods to build integrations that require strong typecasting. The downside of the Enterprise WSDL is that it only works with the schema of a single Salesforce organization because it is bound to all of the unique objects and fields that exist in that organization's data model.

Entity Relationship Diagram (ERD)

A data modeling tool that helps you organize your data into entities (or objects, as they are called in the Force.com platform) and define the relationships between them. ERD diagrams for key Salesforce objects are published in the [SOAP API Developer's Guide](#).

F

Field

A part of an object that holds a specific piece of information, such as a text or currency value.

Field-Level Security

Settings that determine whether fields are hidden, visible, read only, or editable for users. Available in Professional, Enterprise, Unlimited, Performance, and Developer Editions.

Filter Condition/Criteria

Condition on particular fields that qualifies items to be included in a list view or report, such as "State equals California."

Force.com

The Salesforce platform for building applications in the cloud. Force.com combines a powerful user interface, operating system, and database to allow you to customize and deploy applications in the cloud for your entire enterprise.

Foreign Key

A field whose value is the same as the primary key of another table. You can think of a foreign key as a copy of a primary key from another table. A relationship is made between two tables by matching the values of the foreign key in one table with the values of the primary key in another.

Formula Field

A type of custom field. Formula fields automatically calculate their values based on the values of merge fields, expressions, or other values.

Function

Built-in formulas that you can customize with input parameters. For example, the DATE function creates a date field type from a given year, month, and day.

G

Gregorian Year

A calendar based on a 12-month structure used throughout much of the world.

Group Edition

A product designed for small businesses and workgroups with a limited number of users.

H

HTTP Debugger

An application that can be used to identify and inspect SOAP requests that are sent from the AJAX Toolkit. They behave as proxy servers running on your local machine and allow you to inspect and author individual requests.

I

ID

See Salesforce Record ID.

Inline S-Control

 **Note:** S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won't be allowed to create them. Existing s-controls will remain unaffected, and can still be edited.

An s-control that displays within a record detail page or dashboard, rather than on its own page.

Instance

The cluster of software and hardware represented as a single logical server that hosts an organization's data and runs their applications. The Force.com platform runs on multiple instances, but data for any single organization is always stored on a single instance.

Integration User

A Salesforce user defined solely for client apps or integrations. Also referred to as the logged-in user in a SOAP API context.

ISO Code

The International Organization for Standardization country code, which represents each country by two letters.

J

Junction Object

A custom object with two master-detail relationships. Using a custom junction object, you can model a "many-to-many" relationship between two objects. For example, you may have a custom object called "Bug" that relates to the standard case object such that a bug could be related to multiple cases and a case could also be related to multiple bugs.

K

No Glossary items for this entry.

L

License Management Application (LMA)

A free AppExchange app that allows you to track sales leads and accounts for every user who downloads your managed package (app) from the AppExchange.

License Management Organization (LMO)

The Salesforce organization that you use to track all the Salesforce users who install your package. A license management organization must have the License Management Application (LMA) installed. It automatically receives notification every time your package is installed or uninstalled so that you can easily notify users of upgrades. You can specify any Enterprise, Unlimited, Performance, or Developer Edition organization as your license management organization. For more information, go to <http://www.salesforce.com/docs/en/lma/index.htm>.

List View

A list display of items (for example, accounts or contacts) based on specific criteria. Salesforce provides some predefined views.

In the Agent console, the list view is the top frame that displays a list view of records based on specific criteria. The list views you can select to display in the console are the same list views defined on the tabs of other objects. You cannot create a list view within the console.

Locale

The country or geographic region in which the user is located. The setting affects the format of date and number fields, for example, dates in the English (United States) locale display as 06/30/2000 and as 30/06/2000 in the English (United Kingdom) locale.

In Professional, Enterprise, Unlimited, Performance, and Developer Edition organizations, a user's individual `Locale` setting overrides the organization's `Default Locale` setting. In Personal and Group Editions, the organization-level locale field is called `Locale`, not `Default Locale`.

Logged-in User

In a SOAP API context, the username used to log into Salesforce. Client applications run with the permissions and sharing of the logged-in user. Also referred to as an integration user.

M

Managed Package

A collection of application components that is posted as a unit on the AppExchange and associated with a namespace and possibly a License Management Organization. To support upgrades, a package must be managed. An organization can create a single managed package that can be downloaded and installed by many different organizations. Managed packages differ from unmanaged packages by having some locked components, allowing the managed package to be upgraded later. Unmanaged packages do not include locked components and cannot be upgraded. In addition, managed packages obfuscate certain components (like Apex) on subscribing organizations to protect the intellectual property of the developer.

Manual Sharing

Record-level access rules that allow record owners to give read and edit permissions to other users who might not have access to the record any other way.

Glossary

Many-to-Many Relationship

A relationship where each side of the relationship can have many children on the other side. Many-to-many relationships are implemented through the use of junction objects.

Master-Detail Relationship

A relationship between two different types of records that associates the records with each other. For example, accounts have a master-detail relationship with opportunities. This type of relationship affects record deletion, security, and makes the lookup relationship field required on the page layout.

Master Picklist

A complete list of picklist values available for a record type or business process.

Metadata

Information about the structure, appearance, and functionality of an organization and any of its parts. Force.com uses XML to describe metadata.

Metadata WSDL

A WSDL for users who want to use the Force.com Metadata API calls.

Multitenancy

An application model where all users and apps share a single, common infrastructure and code base.

N

Namespace

In a packaging context, a one- to 15-character alphanumeric identifier that distinguishes your package and its contents from packages of other developers on AppExchange, similar to a domain name. Salesforce automatically prepends your namespace prefix, followed by two underscores ("__"), to all unique component names in your Salesforce organization.

Native App

An app that is built exclusively with setup (metadata) configuration on Force.com. Native apps do not require any external services or infrastructure.

O

Object

An object allows you to store information in your Salesforce organization. The object is the overall definition of the type of information you are storing. For example, the case object allows you to store information regarding customer inquiries. For each object, your organization will have multiple records that store the information about specific instances of that type of data. For example, you might have a case record to store the information about Joe Smith's training inquiry and another case record to store the information about Mary Johnson's configuration issue.

Object-Level Help

Custom help text that you can provide for any custom object. It displays on custom object record home (overview), detail, and edit pages, as well as list views and related lists.

Object-Level Security

Settings that allow an administrator to hide whole objects from users so that they don't know that type of data exists. Object-level security is specified with object permissions.

onClick JavaScript

JavaScript code that executes when a button or link is clicked.

Glossary

One-to-Many Relationship

A relationship in which a single object is related to many other objects. For example, an account may have one or more related contacts.

Organization-Wide Defaults

Settings that allow you to specify the baseline level of data access that a user has in your organization. For example, you can set organization-wide defaults so that any user can see any record of a particular object that is enabled via their object permissions, but they need extra permissions to edit one.

Outbound Call

Any call that originates from a user to a number outside of a call center in Salesforce CRM Call Center.

Outbound Message

An outbound message sends information to a designated endpoint, like an external service. Outbound messages are configured from Setup. You must configure the external endpoint and create a listener for the messages using the SOAP API.

Overlay

An overlay displays additional information when you hover your mouse over certain user interface elements. Depending on the overlay, it will close when you move your mouse away, click outside of the overlay, or click a close button.

Owner

Individual user to which a record (for example, a contact or case) is assigned.

P

PaaS

See Platform as a Service.

Package

A group of Force.com components and applications that are made available to other organizations through the AppExchange. You use packages to bundle an app along with any related components so that you can upload them to AppExchange together.

Package Dependency

This is created when one component references another component, permission, or preference that is required for the component to be valid. Components can include but are not limited to:

- Standard or custom fields
- Standard or custom objects
- Visualforce pages
- Apex code

Permissions and preferences can include but are not limited to:

- Divisions
- Multicurrency
- Record types

Package Installation

Installation incorporates the contents of a package into your Salesforce organization. A package on the AppExchange can include an app, a component, or a combination of the two. After you install a package, you may need to deploy components in the package to make it generally available to the users in your organization.

Package Publication

Publishing your package makes it publicly available on the AppExchange.

Package Version

A package version is a number that identifies the set of components uploaded in a package. The version number has the format *majorNumber.minorNumber.patchNumber* (for example, 2.1.3). The major and minor numbers increase to a chosen value during every major release. The *patchNumber* is generated and updated only for a patch release.

Unmanaged packages are not upgradeable, so each package version is simply a set of components for distribution. A package version has more significance for managed packages. Packages can exhibit different behavior for different versions. Publishers can use package versions to evolve the components in their managed packages gracefully by releasing subsequent package versions without breaking existing customer integrations using the package. See also Patch and Patch Development Organization.

Parent Account

An organization or company that an account is affiliated. By specifying a parent for an account, you can get a global view of all parent/subsidiary relationships using the [View Hierarchy](#) link.

Partner WSDL

A loosely-typed WSDL for customers, partners, and ISVs who want to build an integration or an AppExchange app that can work across multiple Salesforce organizations. With this WSDL, the developer is responsible for marshaling data in the correct object representation, which typically involves editing the XML. However, the developer is also freed from being dependent on any particular data model or Salesforce organization. Contrast this with the Enterprise WSDL, which is strongly typed.

Patch

A patch enables a developer to change the functionality of existing components in a managed package, while ensuring subscribing organizations that there are no visible behavior changes to the package. For example, you can add new variables or change the body of an Apex class, but you may not add, deprecate, or remove any of its methods. Patches are tracked by a *patchNumber* appended to every package version. See also Patch Development Organization and Package Version.

Patch Development Organization

The organization where patch versions are developed, maintained, and uploaded. Patch development organizations are created automatically for a developer organization when they request to create a patch. See also Patch and Package Version.

Personal Edition

Product designed for individual sales representatives and single users.

Personal Information

User information including personal contact information, quotas, personal group information, and default opportunity team.

Picklist

Selection list of options available for specific fields in a Salesforce object, for example, the `Industry` field for accounts. Users can choose a single value from a list of options rather than make an entry directly in the field. See also Master Picklist.

Picklist (Multi-Select)

Selection list of options available for specific fields in a Salesforce object. Multi-select picklists allow users to choose one or more values. Users can choose a value by double clicking on it, or choose additional values from a scrolling list by holding down the CTRL key while clicking a value and using the arrow icon to move them to the selected box.

Picklist Values

Selections displayed in drop-down lists for particular fields. Some values come predefined, and other values can be changed or defined by an administrator.

Platform as a Service (PaaS)

An environment where developers use programming tools offered by a service provider to create applications and deploy them in a cloud. The application is hosted as a service and provided to customers via the Internet. The PaaS vendor provides an API for creating and extending specialized applications. The PaaS vendor also takes responsibility for the daily maintenance, operation, and support of the deployed application and each customer's data. The service alleviates the need for programmers to install, configure, and maintain the applications on their own hardware, software, and related IT resources. Services can be delivered using the PaaS environment to any market segment.

Glossary

Platform Edition

A Salesforce edition based on Enterprise, Unlimited, or Performance Edition that does not include any of the standard Salesforce apps, such as Sales or Service & Support.

Primary Key

A relational database concept. Each table in a relational database has a field in which the data value uniquely identifies the record. This field is called the primary key. The relationship is made between two tables by matching the values of the foreign key in one table with the values of the primary key in another.

Production Organization

A Salesforce organization that has live users accessing data.

Professional Edition

A Salesforce edition designed for businesses who need full-featured CRM functionality.

Q

Queue

A holding area for items before they are processed. Salesforce uses queues in a number of different features and technologies.

Query Locator

A parameter returned from the `query()` or `queryMore()` API call that specifies the index of the last result record that was returned.

Query String Parameter

A name-value pair that's included in a URL, typically after a '?' character. For example:

```
https://yourInstance.salesforce.com/001/e?name=value
```

R

Record

A single instance of a Salesforce object. For example, "John Jones" might be the name of a contact record.

Record Name

A standard field on all Salesforce objects. Whenever a record name is displayed in a Force.com application, the value is represented as a link to a detail view of the record. A record name can be either free-form text or an autonumber field. Record Name does not have to be a unique value.

Record Type

A record type is a field available for certain records that can include some or all of the standard and custom picklist values for that record. You can associate record types with profiles to make only the included picklist values available to users with that profile.

Record-Level Security

A method of controlling data in which you can allow a particular user to view and edit an object, but then restrict the records that the user is allowed to see.

Recycle Bin

A page that lets you view and restore deleted information. Access the Recycle Bin by using the link in the sidebar.

Related Object

Objects chosen by an administrator to display in the Agent console's mini view when records of a particular type are shown in the console's detail view. For example, when a case is in the detail view, an administrator can choose to display an associated account, contact, or asset in the mini view.

Relationship

A connection between two objects, used to create related lists in page layouts and detail levels in reports. Matching values in a specified field in both objects are used to link related data; for example, if one object stores data about companies and another object stores data about people, a relationship allows you to find out which people work at the company.

Relationship Query

In a SOQL context, a query that traverses the relationships between objects to identify and return results. Parent-to-child and child-to-parent syntax differs in SOQL queries.

Report Type

A *report type* defines the set of records and fields available to a report based on the relationships between a primary object and its related objects. Reports display only records that meet the criteria defined in the report type. Salesforce provides a set of pre-defined standard report types; administrators can create custom report types as well.

Role Hierarchy

A record-level security setting that defines different levels of users such that users at higher levels can view and edit information owned by or shared with users beneath them in the role hierarchy, regardless of the organization-wide sharing model settings.

Roll-Up Summary Field

A field type that automatically provides aggregate values from child records in a master-detail relationship.

Running User

Each dashboard has a *running user*, whose security settings determine which data to display in a dashboard. If the running user is a specific user, all dashboard viewers see data based on the security settings of that user—regardless of their own personal security settings. For dynamic dashboards, you can set the running user to be the logged-in user, so that each user sees the dashboard according to his or her own access level.

S

SaaS

See Software as a Service (SaaS).

S-Control

 **Note:** S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won't be allowed to create them. Existing s-controls will remain unaffected, and can still be edited.

Custom Web content for use in custom links. Custom s-controls can contain any type of content that you can display in a browser, for example a Java applet, an Active-X control, an Excel file, or a custom HTML Web form.

Salesforce Record ID

A unique 15- or 18-character alphanumeric string that identifies a single record in Salesforce.

Salesforce SOA (Service-Oriented Architecture)

A powerful capability of Force.com that allows you to make calls to external Web services from within Apex.

Sandbox

A nearly identical copy of a Salesforce production organization for development, testing, and training. The content and size of a sandbox varies depending on the type of sandbox and the editioin of the production organization associated with the sandbox.

Search Layout

The organization of fields included in search results, in lookup dialogs, and in the key lists on tab home pages.

Search Phrase

Search phrases are queries that users enter when searching on www.google.com.

Glossary

Semi-Join

A semi-join is a subquery on another object in an `IN` clause in a SOQL query. You can use semi-joins to create advanced queries, such as getting all contacts for accounts that have an opportunity with a particular record type. See also Anti-Join.

Session ID

An authentication token that is returned when a user successfully logs in to Salesforce. The Session ID prevents a user from having to log in again every time he or she wants to perform another action in Salesforce. Different from a record ID or Salesforce ID, which are terms for the unique ID of a Salesforce record.

Session Timeout

The period of time after login before a user is automatically logged out. Sessions expire automatically after a predetermined length of inactivity, which can be configured in Salesforce from Setup by clicking **Security Controls**. The default is 120 minutes (two hours). The inactivity timer is reset to zero if a user takes an action in the Web interface or makes an API call.

Setup

A menu where administrators can customize and define organization settings and Force.com apps. Depending on your organization's user interface settings, Setup may be a link in the user interface header or in the drop-down list under your name.

Sharing

Allowing other users to view or edit information you own. There are different ways to share data:

- Sharing Model—defines the default organization-wide access levels that users have to each other's information and whether to use the hierarchies when determining access to data.
- Role Hierarchy—defines different levels of users such that users at higher levels can view and edit information owned by or shared with users beneath them in the role hierarchy, regardless of the organization-wide sharing model settings.
- Sharing Rules—allow an administrator to specify that all information created by users within a given group or role is automatically shared to the members of another group or role.
- Manual Sharing—allows individual users to share records with other users or groups.
- Apex-Managed Sharing—enables developers to programmatically manipulate sharing to support their application's behavior. See Apex-Managed Sharing.

Sharing Model

Behavior defined by your administrator that determines default access by users to different types of records.

Sharing Rule

Type of default sharing created by administrators. Allows users in a specified group or role to have access to all information created by users within a given group or role.

Sites

Force.com Sites enables you to create public websites and applications that are directly integrated with your Salesforce organization—without requiring users to log in with a username and password.

Snippet

 **Note:** S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won't be allowed to create them. Existing s-controls will remain unaffected, and can still be edited.

A type of s-control that is designed to be included in other s-controls. Similar to a helper method that is used by other methods in a piece of code, a snippet allows you to maintain a single copy of HTML or JavaScript that you can reuse in multiple s-controls.

SOAP (Simple Object Access Protocol)

A protocol that defines a uniform way of passing XML-encoded data.

SOAP API

A SOAP-based Web services application programming interface that provides access to your Salesforce organization's information.

sObject

The abstract or parent object for all objects that can be stored in the Force.com platform.

Software as a Service (SaaS)

A delivery model where a software application is hosted as a service and provided to customers via the Internet. The SaaS vendor takes responsibility for the daily maintenance, operation, and support of the application and each customer's data. The service alleviates the need for customers to install, configure, and maintain applications with their own hardware, software, and related IT resources. Services can be delivered using the SaaS model to any market segment.

SOQL (Salesforce Object Query Language)

A query language that allows you to construct simple but powerful query strings and to specify the criteria that should be used to select data from the Force.com database.

SOSL (Salesforce Object Search Language)

A query language that allows you to perform text-based searches using the Force.com API.

Standard Object

A built-in object included with the Force.com platform. You can also build custom objects to store information that is unique to your app.

Syndication Feeds

Give users the ability to subscribe to changes within Force.com sites and receive updates in external news readers.

System Log

Part of the Developer Console, a separate window console that can be used for debugging code snippets. Enter the code you want to test at the bottom of the window and click Execute. The body of the System Log displays system resource information, such as how long a line took to execute or how many database calls were made. If the code did not run to completion, the console also displays debugging information.

T

Test Method

An Apex class method that verifies whether a particular piece of code is working properly. Test methods take no arguments, commit no data to the database, and can be executed by the `runTests()` system method either through the command line or in an Apex IDE, such as the Force.com IDE.

Translation Workbench

The Translation Workbench lets you specify languages you want to translate, assign translators to languages, create translations for customizations you've made to your Salesforce organization, and override labels and translations from managed packages. Everything from custom picklist values to custom fields can be translated so your global users can use all of Salesforce in their language.

Trigger

A piece of Apex that executes before or after records of a particular type are inserted, updated, or deleted from the database. Every trigger runs with a set of context variables that provide access to the records that caused the trigger to fire, and all triggers run in bulk mode—that is, they process several records at once, rather than just one record at a time.

Trigger Context Variable

Default variables that provide access to information about the trigger and the records that caused it to fire.

U

Unit Test

A unit is the smallest testable part of an application, usually a method. A unit test operates on that piece of code to make sure it works correctly. See also Test Method.

Unlimited Edition

Unlimited Edition is Salesforce's solution for maximizing your success and extending that success across the entire enterprise through the Force.com platform.

Unmanaged Package

A package that cannot be upgraded or controlled by its developer.

URL (Uniform Resource Locator)

The global address of a website, document, or other resource on the Internet. For example, <http://www.salesforce.com>.

URL S-Control

 **Note:** S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won't be allowed to create them. Existing s-controls will remain unaffected, and can still be edited.

An s-control that contains an external URL that hosts the HTML that should be rendered on a page. When saved this way, the HTML is hosted and run by an external website. URL s-controls are also called Web controls.

V

Validation Rule

A rule that prevents a record from being saved if it does not meet the standards that are specified.

Visualforce

A simple, tag-based markup language that allows developers to easily define custom pages and components for apps built on the platform. Each tag corresponds to a coarse or fine-grained component, such as a section of a page, a related list, or a field. The components can either be controlled by the same logic that is used in standard Salesforce pages, or developers can associate their own logic with a controller written in Apex.

W

Web Control

See URL S-Control.

Web Links

See Custom Links.

Web Service

A mechanism by which two applications can easily exchange data over the Internet, even if they run on different platforms, are written in different languages, or are geographically remote from each other.

Web Services API

A Web services application programming interface that provides access to your Salesforce organization's information. See also SOAP API and Bulk API.

Glossary

WebService Method

An Apex class method or variable that can be used by external systems, like a mash-up with a third-party application. Web service methods must be defined in a global class.

Web Tab

A custom tab that allows your users to use external websites from within the application.

Automated Actions

Automated actions, such as email alerts, tasks, field updates, and outbound messages, can be triggered by a process, workflow rule, approval process, or milestone.

Workflow Action

A workflow action, such as an email alert, field update, outbound message, or task, fires when the conditions of a workflow rule are met.

Workflow Email Alert

A workflow action that sends an email when a workflow rule is triggered. Unlike workflow tasks, which can only be assigned to application users, workflow alerts can be sent to any user or contact, as long as they have a valid email address.

Workflow Field Update

A workflow action that changes the value of a particular field on a record when a workflow rule is triggered.

Workflow Outbound Message

A workflow action that sends data to an external Web service, such as another cloud computing application. Outbound messages are used primarily with composite apps.

Workflow Queue

A list of workflow actions that are scheduled to fire based on workflow rules that have one or more time-dependent workflow actions.

Workflow Rule

A workflow rule sets workflow actions into motion when its designated conditions are met. You can configure workflow actions to execute immediately when a record meets the conditions in your workflow rule, or set time triggers that execute the workflow actions on a specific day.

Workflow Task

A workflow action that assigns a task to an application user when a workflow rule is triggered.

Wrapper Class

A class that abstracts common functions such as logging in, managing sessions, and querying and batching records. A wrapper class makes an integration more straightforward to develop and maintain, keeps program logic in one place, and affords easy reuse across components. Examples of wrapper classes in Salesforce include the AJAX Toolkit, which is a JavaScript wrapper around the Salesforce SOAP API, wrapper classes such as `CCritical_Section` in the CTI Adapter for Salesforce CRM Call Center, or wrapper classes created as part of a client integration application that accesses Salesforce using the SOAP API.

WSC (Web Service Connector)

An XML-based Web service framework that consists of a Java implementation of a SOAP server. With WSC, developers can develop client applications in Java by using Java classes generated from Salesforce Enterprise WSDL or Partner WSDL.

WSDL (Web Services Description Language) File

An XML file that describes the format of messages you send and receive from a Web service. Your development environment's SOAP client uses the Salesforce Enterprise WSDL or Partner WSDL to communicate with Salesforce using the SOAP API.

X

No Glossary items for this entry.

Glossary

Y

No Glossary items for this entry.

Z

No Glossary items for this entry.

INDEX

.NET handling of non-string fields [2382](#)

A

AcceptedEventRelation object [168](#)
Account object [169](#)
AccountCleanInfo object [191](#)
AccountContactRelation object [189](#)
AccountContactRole object [213](#)
AccountFeed object [214](#)
AccountHistory object [222](#)
AccountOwnerSharingRule object [223](#)
AccountPartner object [226](#)
AccountShare object [228](#)
AccountTag object [232](#)
AccountTeamMember object [233](#)
AccountTerritoryAssignmentRule object [237](#)
AccountTerritoryAssignmentRuleItem object [238](#)
ActionLinkGroupTemplate object [243](#)
ActionLinkTemplate object [245](#)
ActivityHistory [250](#)
AdditionalInformationMap [68](#)
AdditionalNumber object [258](#)
address [38, 42](#)
AgentWork object [259](#)
AggregateResult object [2204](#)
AllOrNoneHeader header [2356](#)
AllowedEmailDomain object [265](#)
AllowFieldTruncationHeader header [2357](#)
Announcement object [266](#)
ApexClass object [268](#)
ApexComponent object [270](#)
ApexLog object [273](#)
ApexPage object [275](#)
ApexTestQueueItem object [278](#)
ApexTestResult object [280](#)
ApexTestResultLimits object [283](#)
ApexTestRunResult object [286](#)
ApexTestSuite object [289](#)
ApexTrigger object [289](#)
API
 Bulk API [2393](#)
 choosing [2393](#)
API usage metering [2378](#)
AppMenuItem object [293](#)
Approval object [299](#)

Article

 DataCategorySelection [301](#)
Asset object [309](#)
AssetFeed object [313](#)
AssetTag object [324](#)
AssignedResource object [326](#)
AssignmentRule object [327](#)
AssignmentRuleHeader header [2359](#)
AsyncApexJob object [328](#)
AttachedContentDocument object [331](#)
AttachedContentNote object [333](#)
Attachment object [335](#)
AuraDefinition object [339](#)
AuraDefinitionBundle object [341](#)
AuthConfig object [343](#)
AuthConfigProviders object [346](#)
Authentication [86](#)
AuthProvider object [347](#)
AuthSession object [352](#)

B

BackgroundOperation object [355](#)
Badge
 data model [129](#)
Bookmark object [362](#)
BrandTemplate [363](#)
BusinessProcess object [369](#)

C

Call centers and the API [2406](#)
CallCenter object [371](#)
CallOptions header [2360](#)
Calls
 convertLead() [2142](#)
 create() [2149](#)
 delete() [2159](#)
 describeAllTabs() [2237](#)
 describeAppMenu() [2238](#)
 describeApprovalLayout() [2240](#)
 describeAvailableQuickActions() [2242](#)
 describeCompactLayouts() [2244](#)
 describeDataCategoryGroups() [2216, 2219, 2247, 2249–2250, 2254](#)
 describeFlexiPages() [2254](#)
 describeGlobal() [2259](#)
 describeGlobalTheme() [2263](#)

Calls (*continued*)

- describeKnowledge() 2265
- describeLayout() 2266
- describePrimaryCompactLayouts() 2283
- describeQuickActions() 2285
- describeSearchLayouts() 2292
- describeSearchScopeOrder() 2291
- describeSObject() 2294
- describeSObjects() 2297–2298
- describeSoftphoneLayout() 2314
- describeSoqlListViews() 2318
- describeTabs() 2321
- describeTheme() 2326
- emptyRecycleBin() 2162
- executeListView() 2165
- getDeleted() 2172
- getServerTimestamp() 2329
- getUpdated() 2176
- getUserInfo() 2331
- invalidateSessions() 2180
- list of core calls 2141
- list of describe calls 2236
- list of utility calls 2329
- login() 2182
- logout() 2188
- merge() 2189
- performQuickActions() 2195
- process() 2196
- query() 2200
- queryAll() 2206
- queryMore() 2209
- resetPassword() 2338
- retrieve() 2213
- sendEmail() 2334, 2340
- sendEmailMessage 2350
- setPassword() 2353
- undelete() 2219
- update() 2223
- upsert() 2230
- Campaign object 373
- CampaignFeed object 381
- CampaignInfluence object 389
- CampaignInfluenceModel object 390
- CampaignMember object 392
- CampaignMemberStatus object 399
- CampaignOwnerSharingRule object 401
- CampaignShare object 403
- CampaignTag object 404
- Cascading deletes 71
- Case object 406
- CaseComment object 417
- CaseContactRole object 420
- CaseFeed object 421
- CaseHistory object 429
- CaseMilestone object 430
- CaseOwnerSharingRule object 434
- CaseShare object 436
- CaseSolution object 438
- CaseStatus object 439
- CaseTag object 440
- CaseTeamMember object 442
- CaseTeamRole object 443
- CaseTeamTemplate object 443
- CaseTeamTemplateMember object 444
- CaseTeamTemplateRecord object 445
- CategoryData object 446
- CategoryNode object 447
- CategoryNodeLocalization object 448
- Chatter
- objects 125
- ChatterActivity object 453
- ChatterAnswersActivity object 455
- ChatterAnswersReputationLevel object 459
- ChatterConversation object 460
- ChatterConversationMember object 461
- ChatterMessage object 462
- Choosing an API 2393
- Chunking of HTTP 2381
- Cipher key length 86
- Client certificate download 2387
- Client parameter in CallOptions header 2360
- Client parameter in OwnerChangeOptions header 2369
- ClientBrowser object 463
- CollaborationGroup object 465
- CollaborationGroupFeed object 471
- CollaborationGroupMember object 478
- CollaborationGroupMemberRequest object 480
- CollaborationGroupRecord object 482
- CollaborationInvitation object 483
- CombinedAttachment object 486
- Community (Zone) object 488
- compound fields 38, 40, 42
- Compression 2380
- ConnectedApplication object 490
- Connections, HTTP persistent 2381
- Contact object 492
- ContactCleanInfo object 504
- ContactFeed object 514

- ContactHistory object [521](#)
ContactOwnerSharingRule object [522](#)
ContactShare object [524](#)
ContactTag object [526](#)
Content type requirements [2378](#)
ContentDistribution object [529](#)
ContentDistributionView object [535](#)
ContentDocument object [536](#)
ContentDocumentFeed object [542](#)
ContentDocumentHistory object [549](#)
ContentDocumentLink object [551](#)
ContentFolder object [555](#)
ContentFolderItem object [555](#)
ContentFolderLink object [557](#)
ContentFolderMember object [558](#)
ContentHubItem object [559](#)
ContentHubRepository object [563](#)
ContentNote object [564](#)
ContentVersion object [568](#)
ContentVersionHistory object [579](#)
ContentWorkspace object [581](#)
ContentWorkspaceDoc object [582](#)
Contract object [584](#)
ContractContactRole object [593](#)
ContractFeed object [594](#)
ContractHistory object [602](#)
ContractLineItem object [603](#)
ContractLineItemHistory object [607](#)
ContractStatus object [609](#)
ContractTag object [611](#)
convertLead() call [2142](#)
create() call [2149](#)
CreatedByld fields [68](#)
CreatedDate fields [68](#)
CronJobDetail object [612](#)
CronTrigger object [613](#)
CSS for Salesforce look and feel [95](#)
CurrencyType object [616](#)
Custom Metadta Type__mdt object [621](#)
Custom Object__Feed object [623](#)
Custom objects [73](#)
CustomBrand object [617](#)
CustomBrandAsset object [618](#)
CustomPermission object [630](#)
CustomPermissionDependency object [633](#)
- D**
- DandBCompany object [635](#)
Dashboard object [658](#)
- DashboardComponent object [663](#)
DashboardComponentFeed object [664](#)
DashboardFeed object [672](#)
DashboardTag object [679](#)
Data types, mapping API to Salesforce field types [43](#)
DataAssessmentFieldMetric object [680](#)
DataAssessmentMetric object [682](#)
DataAssessmentValueMetric object [683](#)
Database server instances [2378](#)
DataCategorySelection article [301](#)
DataCategorySelection question [1560](#)
DatacloudCompany object [684](#)
DatacloudContact object [693](#)
DatacloudDandBCompany object [698](#)
DatacloudOwnedEntity object [721](#)
DatacloudPurchaseUsage object [723](#)
DatacloudSocialHandle object [725](#)
DatedConversionRate object [726](#)
DcSocialProfile object [727](#)
DcSocialProfileHandle object [729](#)
DeclinedEventRelation object [730](#)
delete() call [2159](#)
Deleting and cascading deletes [71](#)
describeAllTabs() call [2237](#)
describeAppMenu() call [2238](#)
describeApprovalLayout()
 DescribeApprovalLayoutResult [2242](#)
describeApprovalLayout() call [2240](#)
DescribeApprovalLayoutResult [2242](#)
DescribeAvailableQuickActionResult [2243](#)
describeAvailableQuickActions() call [2242](#)
DescribeColor [2324](#)
describeCompactLayouts()
 DescribeCompactLayoutsResult [2246](#)
describeCompactLayouts() call [2244](#)
DescribeCompactLayoutsResult [2246](#)
describeDataCategoryGroups() call [2216, 2219, 2247, 2249–2250, 2254](#)
describeFlexiPages() call [2254](#)
describeGlobal() call [2259](#)
DescribeGlobalTheme [2264](#)
describeGlobalTheme() call [2263](#)
Describelcon [2325](#)
describeKnowledge() call [2265](#)
describeLayout() call [2266](#)
describePrimaryCompactLayouts() call [2283](#)
DescribeQuickActionResult [2286](#)
describeQuickActions()
 DescribeAvailableQuickActionResult [2243](#)

describeQuickActions() (continued)
 DescribeQuickActionResult 2286
 describeQuickActions() call 2285
 DescribeSearchLayoutResult 2292, 2294
 describeSearchLayouts()
 DescribeSearchLayoutResult 2294
 describeSearchLayouts() call 2292
 describeSearchScopeOrder() call 2291
 describeSObject() call 2294
 describeSObjects() call 2297–2298
 describeSoftphoneLayout() call 2314
 describeSqlListViews() call 2318
 describeTabs() call 2321
 describeTheme() call 2326
 DescribeThemeItem 2328
 DescribeThemeResult 2327
 DisableFeedTrackingHeader header 2361
 Division object 732
 DivisionLocalization object 733
 Document object 735
 DocumentAttachmentMap 739
 DocumentTag object 740
 Domain object 742
 DomainSite object 743
 Duplicate Rule 2362
 DuplicateError 60
 DuplicateResult 61
 DuplicateRule 747
 DuplicateRuleHeader 2362

E

Effective dated currency 726
 EmailDomainKey object 749
 EmailHeader header 2363
 EmailMessage object 751
 EmailMessageRelation object 757
 EmailServicesAddress object 759
 EmailServicesFunction object 761
 EmailStatus 766
 EmailTemplate object 739, 767
 emptyRecycleBin() call 2162
 Enterprise WSDL and .NET 2382
 Entitlement object 772
 EntitlementContact object 776
 EntitlementFeed object 777
 EntitlementHistory object 785
 EntitlementTemplate object 786
 EntityHistory object 788
 EntityMilestone object 790

EntitySubscription object 797
 EnvironmentHubMember object 799
 Error handling 83
 Event object 803
 EventFeed object 817
 EventLogFile 824
 EventRelation object 948
 EventTag object 953
 EventWhoRelation object 954
 executeListView() call 2165
 Expiration of session ID 84
 External ID fields 2230
 External objects 76
 External Objects
 queryAll() limitations 2406
 queryMore() limitations 2406
 ExternalDataSource object 955
 ExternalDataUserAuth object 958

F

FeedAttachment object 960
 Feedback
 data model 130
 FeedComment object 963
 FeedItem object 968, 985
 FeedLike object 978
 FeedPollChoice object 979
 FeedPollVote object 981
 FeedPost object 981
 Feeds
 custom 988
 post 981, 988
 standard 988
 FeedTrackedChange object 988
 Field types 32, 38, 40, 42
 FieldDiff 67
 FieldHistoryArchive object 360, 991
 FieldPermissions object 994
 Fields
 .NET and the enterprise WSDL handling of non-string fields
 2382
 external ID fields 2230
 system fields 68
 FiscalYearSettings object 997
 FlowInterview object 1002
 Folder object 1006
 FolderedContentDocument object 1009
 ForecastingAdjustment object 1010
 ForecastingFact object 1015

Index

ForecastingItem object 1017
ForecastingOwnerAdjustment object 1023
ForecastingQuota object 1027
ForecastingType object 1030
ForecastShare object 1032
Freeze users 1932

G

geolocation 38, 40, 42
getDeleted() call 2172
getServerTimestamp() call 2329
getUpdated() call 2176
getUserInfo() call 2331
Goal object 1033
GoalFeed object 1036
GoalHistory object 1041
GoalLink object 1042
GoalShare object 1043
GoogleDoc object 1045
GROUP BY 2204
Group object 1046
GroupMember object 1049

H

HashtagDefinition object 1050
Headers
 AllOrNoneHeader 2356
 AllowFieldTruncationHeader 2357
 AssignmentRuleHeader 2359
 CallOptions 2360
 DisableFeedTrackingHeader 2361
 DuplicateRuleHeader 2362
 EmailHeader 2363
 LimitInfoHeader 2365
 LocaleOptions 2366
 LoginScopeHeader 2367
 MruHeader 2368
 OwnerChangeOptions 2369
 PackageVersionHeader 2372
 QueryOptions 2373
 SessionHeader 2374
 SOAP 2355
 UserTerritoryDeleteHeader 2374
Holiday object 1052
HTTP chunking 2381
HTTP persistent connections 2381

|
ID fields 68

Idea object 1055
IdeaComment object 1061
IdeaReputation 1063
IdeaReputationLevel object 1066
IdeaTheme object 1067
Implicit security restrictions for objects and field 88
Internationalization 2381
invalidateSessions() call 2180
ISO-8859-1 2381

K

KnowledgeableUser object 1069

L

LastModifiedByld fields 68
LastModifiedDate fields 68
Lead object 1085
LeadCleanInfo object 1103
LeadFeed object 1117
LeadHistory object 1124
LeadOwnerSharingRule object 1126
LeadShare object 1128
LeadStatus object 1130
LeadTag object 1131
LimitAllocationPerApp object 1133
LimitInfoHeader header 2365
Limiting API requests (API usage metering) 2378
LineItemOverride object 1135
ListView object 1143
ListViewChart object 1146
ListViewChartInstance object 1147
ListViewColumn object 2167, 2320
LiveAgentSession object 1151
LiveAgentSessionHistory object 1154
LiveAgentSessionOwnerSharingRule object 1156
LiveAgentSessionShare object 1157
LiveChatBlockingRule object 1159
LiveChatButton object 1161
LiveChatButtonDeployment object 1168
LiveChatButtonSkill object 1169
LiveChatDeployment object 1170
LiveChatSensitiveDataRule object 1173
LiveChatTranscript object 1176
LiveChatTranscriptEvent object 1183
LiveChatTranscriptHistory object 1185
LiveChatTranscriptOwnerSharingRule 1186
LiveChatTranscriptOwnerSharingRule object
 LiveChatTranscriptOwnerSharingRule 1186
LiveChatTranscriptShare object 1188

LiveChatTranscriptSkill object 1189
 LiveChatUserConfig object 1190
 LiveChatUserConfigProfile object 1197
 LiveChatUserConfigUser object 1198
 LiveChatVisitor object 1198
 LocaleOptions header 2366
 Localization and character sets 2381
 location 38, 40
 login 86
 login server URL 2376
 login() call
 username 2376
 LoginEvent object 1199
 LoginGeo object 1207
 LoginHistory object 1209
 LoginIp object 1214
 LoginScopeHeader 2367
 logout() call 2188
 LookedUpFromActivity object 1215
 Lookup relationships 71

M

macro 1222
 MacroInstruction object 1222
 MailmergeTemplate object 1224
 Managed packages
 version settings 81
 map 38, 40
 Mapping API data types to Salesforce field types 43
 Master-detail relationships 71
 MatchingRule object 1226
 MatchingRuleItem object 1229
 MatchRecord 66
 MatchResult 64
 merge() call 2189
 MetadataPackage object 1230
 MetadataPackageVersion object 1231
 Metering API usage 2378
 Metric object 1234
 MetricDataLink object 1238
 MetricDataLinkHistory object 1239
 MetricFeed object 1240
 MetricHistory object 1245
 MetricsDataFile object 1246
 MetricShare object 1249
 MilestoneType object 1251
 mobile 38, 40
 MruHeader header 2368

N

NamedCredential object 1255
 NavigationLinkSet object 1258
 NavigationMenuItem object 1259
 NavigationMenuItemLocalization object 1261
 Network object 1263
 NetworkActivityAudit object 1271
 NetworkMember object 1273
 NetworkMemberGroup object 1279
 NetworkModeration object 1280
 NetworkPageOverride object 1282
 NetworkSelfRegistration object 1283
 NewsFeed object 1284
 Note object 1291
 NoteAndAttachment object 1295
 NoteTag object 1296
 Notifications 2385
 notifications() call in outbound messaging WSDL 2388

O

OAuthToken object 1292
 Object_asset 321, 1003
 Object_Assets 323, 1005
 Object_name object 528
 Object[PushTopic] 1541
 Object[StreamingChannel] 1798
 ObjectPermissions object 1298
 Objects
 AcceptedEventRelation 168
 Account 169
 AccountCleanInfo 191
 AccountContactRelation 189
 AccountContractRole 213
 AccountFeed 214
 AccountHistory 222
 AccountOwnerSharingRule 223
 AccountPartner 226
 AccountShare 228
 AccountTag 232
 AccountTeamMember 233
 AccountTerritoryAssignmentRule 237
 AccountTerritoryAssignmentRuleItem 238
 ActionLinkGroupTemplate 243
 ActionLinkTemplate 245
 ActivityHistory 250
 AdditionalInformationMap 68
 AdditionalNumber 258
 AgentWork 259
 AggregateResult 2204

Objects (*continued*)

AllowedEmailDomain 265
 Announcement 266
 ApexClass 268
 ApexComponent 270
 ApexLog 273
 ApexPage 275
 ApexTestResultLimits 283
 ApexTestQueueItem 278
 ApexTestResult 280
 ApexTestRunResult 286
 ApexTestSuite 289
 ApexTrigger 289
 AppMenuItem 293
 Approval 299
 Asset 309
 AssetFeed 313
 Assets 321, 323, 1003, 1005
 AssetTag 324
 AssignedResource 326
 AssignmentRule 327
 AsyncApexJob 328
 AttachedContentDocument 331
 AttachedContentNote 333
 Attachment 335
 AuraDefinition 339
 AuraDefinitionBundle 341
 AuthConfig 343
 AuthConfigProviders 346
 AuthProvider 347
 AuthSession 352
 BackgroundOperation 355
 Bookmark 362
 BrandTemplate 363
 BusinessProcess 369
 CallCenter 371
 Campaign 373
 CampaignFeed 381
 CampaignInfluence 389
 CampaignInfluenceModel 390
 CampaignMember 392
 CampaignMemberStatus 399
 CampaignOwnerSharingRule 401
 CampaignShare 403
 CampaignTag 404
 Case 406
 CaseComment 417
 CaseContactRole 420
 CaseFeed 421

Objects (*continued*)

CaseHistory 429
 CaseMilestone 430
 CaseOwnerSharingRule 434
 CaseShare 436
 CaseSolution 438
 CaseStatus 439
 CaseTag 440
 CaseTeamMember 442
 CaseTeamRole 443
 CaseTeamTemplate 443
 CaseTeamTemplateMember 444
 CaseTeamTemplateRecord 445
 CategoryData 446
 CategoryNode 447
 CategoryNodeLocalization 448
 ChatterActivity 453
 ChatterAnswersActivity 455
 ChatterAnswersReputationLevel 459
 ChatterConversation 460
 ChatterConversationMember 461
 ChatterMessage 462
 ClientBrowser 463
 CollaborationGroup 465
 CollaborationGroupFeed 471
 CollaborationGroupMember 478
 CollaborationGroupMemberRequest 480
 CollaborationGroupRecord 482
 CollaborationInvitation 483
 CombinedAttachment 486
 Community (Zone) 488
 ConnectedApplication 490
 Contact 492
 ContactCleanInfo 504
 ContactFeed 514
 ContactHistory 521
 ContactOwnerSharingRule 522
 ContactShare 524
 ContactTag 526
 ContentDistribution 529
 ContentDistributionView 535
 ContentDocument 536
 ContentDocumentFeed 542
 ContentDocumentHistory 549
 ContentDocumentLink 551
 ContentFolder 555
 ContentFolderItem 555
 ContentFolderLink 557
 ContentFolderMember 558

Objects (*continued*)

[ContentHubItem](#) 559
[ContentHubRepository](#) 563
[ContentNote](#) 564
[ContentVersion](#) 568
[ContentVersionHistory](#) 579
[ContentWorkspace](#) 581
[ContentWorkspaceDoc](#) 582
[Contract](#) 584
[ContractContactRole](#) 593
[ContractFeed](#) 594
[ContractHistory](#) 602
[ContractLineItem](#) 603
[ContractLineItemHistory](#) 607
[ContractStatus](#) 609
[ContractTag](#) 611
[CronJobDetail](#) 612
[CronTrigger](#) 613
[CurrencyType](#) 616
[Custom Metadta Type__mdt](#) 621
[Custom Object__Feed](#) 623
[custom objects](#) 73
[CustomBrand](#) 617
[CustomBrandAsset](#) 618
[CustomPermission](#) 630
[CustomPermissionDependency](#) 633
[DandBCompany](#) 635
[Dashboard](#) 658
[DashboardComponent](#) 663
[DashboardComponentFeed](#) 664
[DashboardFeed](#) 672
[DashboardTag](#) 679
[DataAssessmentFieldMetric](#) 680
[DataAssessmentMetric](#) 682
[DataAssessmentValueMetric](#) 683
[DatacloudCompany](#) 684
[DatacloudContact](#) 693
[DatacloudDandBCompany](#) 698
[DatacloudOwnedEntity](#) 721
[DatacloudPurchaseUsage](#) 723
[DatacloudSocialHandle](#) 725
[DatedConversionRate](#) 726
[DcSocialProfile](#) 727
[DcSocialProfileHandle](#) 729
[DeclinedEventRelation](#) 730
[Division](#) 732
[DivisionLocalization](#) 733
[Document](#) 735
[DocumentAttachmentMap](#) 739

Objects (*continued*)

[DocumentTag](#) 740
[Domain](#) 742
[DomainSite](#) 743
[DuplicateError](#) 60
[DuplicateResult](#) 61
[DuplicateRule](#) 747
[EmailDomainKey](#) 749
[EmailMessage](#) 751
[EmailMessageRelation](#) 757
[EmailServicesAddress](#) 759
[EmailServicesFunction](#) 761
[EmailStatus](#) 766
[EmailTemplate](#) 767
[Entitlement](#) 772
[EntitlementContact](#) 776
[EntitlementFeed](#) 777
[EntitlementHistory](#) 785
[EntitlementTemplate](#) 786
[EntityHistory](#) 788
[EntityMilestone](#) 790
[EntitySubscription](#) 797
[EnvironmentHubMember](#) 799
[Event](#) 803
[EventFeed](#) 817
[EventLogFile](#) 824
[EventRelation](#) 948
[EventTag](#) 953
[EventWhoRelation](#) 954
[external objects](#) 76
[ExternalDataSource](#) 955
[ExternalDataUserAuth](#) 958
[FeedAttachment](#) 960
[FeedComment](#) 963
[FeedItem](#) 968
[FeedLike](#) 978
[FeedPollChoice](#) 979
[FeedPollVote](#) 981
[FeedPost](#) 981
[FeedRevision](#) 985
[FeedTrackedChange](#) 988
[FieldDiff](#) 67
[FieldHistoryArchive](#) 360, 991
[FieldPermissions](#) 994
[FiscalYearSettings](#) 997
[FlowInterview](#) 1002
[Folder](#) 1006
[FolderedContentDocument](#) 1009
[ForecastingAdjustment](#) 1010

Objects (*continued*)

ForecastingFact 1015
 ForecastingItem 1017
 ForecastingOwnerAdjustment 1023
 ForecastingQuota 1027
 ForecastingType 1030
 ForecastShare 1032
 Goal 1033
 GoalFeed 1036
 GoalHistory 1041
 GoalLink 1042
 GoalShare 1043
 GoogleDoc 1045
 Group 1046
 GroupMember 1049
 HashtagDefinition 1050
 Holiday 1052
 Idea 1055
 idea comment 1061
 IdeaReputation 1063
 IdeaReputationLevel 1066
 IdeaTheme 1067
 KnowledgeableUser 1069
 Lead 1085
 LeadCleanInfo 1103
 LeadFeed 1117
 LeadHistory 1124
 LeadOwnerSharingRule 1126
 LeadShare 1128
 LeadStatus 1130
 LeadTag 1131
 LimitAllocationPerApp 1133
 LineItemOverride 1135
 ListView 1143
 ListViewChart 1146
 ListViewChartInstance 1147
 ListViewColumn 2167, 2320
 LiveAgentSession 1151
 LiveAgentSessionHistory 1154
 LiveAgentSessionOwnerSharingRule 1156
 LiveAgentSessionShare 1157
 LiveChatBlockingRule 1159
 LiveChatButton 1161
 LiveChatButtonDeployment 1168
 LiveChatButtonSkill 1169
 LiveChatDeployment 1170
 LiveChatSensitiveDataRule 1173
 LiveChatTranscript 1176
 LiveChatTranscriptEvent 1183

Objects (*continued*)

LiveChatTranscriptHistory 1185
 LiveChatTranscriptShare 1188
 LiveChatTranscriptSkill 1189
 LiveChatUserConfig 1190
 LiveChatUserConfigProfile 1197
 LiveChatUserConfigUser 1198
 LiveChatVisitor 1198
 LoginEvent 1199
 LoginGeo 1207
 LoginHistory 1209
 LoginIp 1214
 LookedUpFromActivity 1215
 MacroInstruction 1222
 MailmergeTemplate 1224
 MatchingRule 1226
 MatchingRuleItem 1229
 MatchRecord 66
 MatchResult 64
 MetadataPackage 1230
 MetadataPackageVersion 1231
 Metric 1234
 MetricDataLink 1238
 MetricDataLinkHistory 1239
 MetricFeed 1240
 MetricHistory 1245
 MetricsDataFile 1246
 MetricShare 1249
 MilestoneType 1251
 NamedCredential 1255
 NavigationLinkSet 1258
 NavigationMenuItem 1259
 NavigationMenuItemLocalization 1261
 Network 1263
 NetworkActivityAudit 1271
 NetworkMember 1273
 NetworkMemberGroup 1279
 NetworkModeration 1280
 NetworkPageOverride 1282
 NetworkSelfRegistration 1283
 NewsFeed 1284
 Note 1291
 NoteAndAttachment 1295
 NoteTag 1296
 OAuthToken 1292
 Object_name 528
 ObjectPermissions 1298
 ObjectTerritory2AssignmentRule 1301
 ObjectTerritory2AssignmentRuleItem 1303

Objects (*continued*)

ObjectTerritory2Association 1304
 OpenActivity 1306
 OperatingHours 1313
 OperatingHoursFeed 1315
 OperatingHoursHistory 1317
 Opportunity 1318
 OpportunityCompetitor 1329
 OpportunityContactRole 1330
 OpportunityFeed 1332
 OpportunityFieldHistory 1339
 OpportunityHistory 1341
 OpportunityLineItem 1343
 OpportunityLineItemSchedule 1349
 OpportunityOverride 1353
 OpportunityOwnerSharingRule 1356
 OpportunityPartner 1358
 OpportunityShare 1360
 OpportunitySplit 1362
 OpportunitySplitType 1364
 OpportunityStage 1367
 OpportunityTag 1369
 OpportunityTeamMember 1370
 Order 1373
 OrderFeed 1383
 OrderHistory 1389
 OrderItem 1391
 OrderItemFeed 1393
 OrderItemHistory 1400
 OrderOwnerSharingRule 1401
 Organization 1404
 OrgWideEmailAddress 1424
 OwnedContentDocument 1425
 PackageLicense 1427
 PackagePushError 1429
 PackagePushJob 1431
 PackagePushRequest 1434
 PackageSubscriber 1436
 Partner 1440
 PartnerNetworkConnection 1443
 PartnerNetworkRecordConnection 1446
 PartnerNetworkSyncLog 1449
 PartnerRole 1451
 Period 1453
 PermissionSet 1455
 PermissionSetAssignment 1461
 PermissionSetLicense 1462
 PermissionSetLicenseAssign 1465
 PlatformAction 1466

Objects (*continued*)

PresenceUserConfig 1473
 PresenceUserConfigProfile 1476
 PresenceUserConfigUser 1477
 Pricebook2 1477
 Pricebook2History 1481
 PricebookEntry 1482
 ProcessDefinition 1485
 ProcessInstance 1487
 ProcessInstanceHistory 1489
 ProcessInstanceNode 1495
 ProcessInstanceStep 1492
 ProcessInstanceWorkitem 1497
 ProcessNode 1499
 Product2 1500
 Product2Feed 1506
 ProductEntitlementTemplate 1514
 Profile 1515
 ProfileSkill 1518
 ProfileSkillEndorsement 1520
 ProfileSkillEndorsementFeed 1521
 ProfileSkillEndorsementHistory 1525
 ProfileSkillFeed 1526
 ProfileSkillHistory 1532
 ProfileSkillShare 1533
 ProfileSkillUser 1534
 ProfileSkillUserFeed 1535
 ProfileSkillUserHistory 1540
 QuantityForecast 1543
 QuantityForecastHistory 1551
 Question 1556
 QuestionReportAbuse 1562
 QuestionSubscription 1563
 QueueRoutingConfig 1553
 QueueSobject 1565
 QuickText 1566
 QuickTextHistory 1568
 QuickTextOwnerSharingRule 1569
 QuickTextShare 1571
 Quote 1572
 QuoteDocument 1585
 QuoteLineItem 1587
 RecordType 1595
 RecordTypeLocalization 1598
 Reply 1599
 ReplyReportAbuse 1602
 Report 1603
 ReportFeed 1607
 ReportTag 1614

Objects (*continued*)

ReputationLevel 1616
 ReputationLevelLocalization 1617
 ReputationPointsRule 1619
 ResourceAbsence 1620
 ResourceAbsenceFeed 1624
 ResourceAbsenceHistory 1626
 RevenueForecast 1627
 RevenueForecastHistory 1636
 RuleTerritory2Association 1638
 SamlSsoConfig 1639
 Scontrol 1644
 ScontrolLocalization 1647
 SecureAgent 1654
 SecureAgentsCluster 1655
 SelfServiceUser 1657
 ServiceAppointment 1660
 ServiceAppointmentFeed 1668
 ServiceAppointmentHistory 1670
 ServiceAppointmentOwnerSharingRule 1671
 ServiceAppointmentShare 1673
 ServiceChannel 1674
 ServiceChannelStatus 1676
 ServiceContract 1677
 ServiceContractFeed 1685
 ServiceContractHistory 1693
 ServiceContractOwnerSharingRule 1694
 ServiceContractShare 1696
 ServicePresenceStatus 1698
 ServiceResource 1699
 ServiceResourceCapacity 1702
 ServiceResourceCapacityFeed 1705
 ServiceResourceCapacityHistory 1707
 ServiceResourceFeed 1708
 ServiceResourceHistory 1711
 ServiceResourceOwnerSharingRule 1712
 ServiceResourceShare 1714
 ServiceResourceSkill 1715
 ServiceResourceSkillFeed 1717
 ServiceResourceSkillHistory 1719
 ServiceTerritory 1720
 ServiceTerritoryFeed 1725
 ServiceTerritoryHistory 1727
 ServiceTerritoryMember 1728
 ServiceTerritoryMemberFeed 1733
 ServiceTerritoryMemberHistory 1735
 SessionPermSetAssignment 1736
 SetupAuditTrail 1738
 SetupEntityAccess 1740

Objects (*continued*)

SignupRequest 1742
 Site 1749
 SiteDomain 1755
 SiteHistory 1756
 Skill 1757
 SkillProfile 1759
 SkillRequirement 1759
 SkillRequirementFeed 1761
 SkillRequirementHistory 1764
 SkillUser 1765
 SlaProcess 1766
 Solution 1769
 SolutionFeed 1773
 SolutionHistory 1780
 SolutionStatus 1782
 SolutionTag 1783
 SOSDeployment 1785
 SOSSession 1787
 SOSSessionActivity 1790
 SOSSessionHistory 1791
 SOSSessionOwnerSharingRule 1792
 SOSSessionShare 1794
 standard objects 131
 StaticResource 1796
 TagDefinition 1800
 Task 1801
 TaskFeed 1812
 TaskPriority 1820
 TaskRelation 1821
 TaskStatus 1823
 TaskTag 1825
 TaskWhoRelation 1826
 TenantSecret 1828
 Territory 1830
 Territory2 1833
 Territory2Model 1835
 Territory2ModelHistory 1837
 Territory2Type 1838
 TestSuiteMembership 1000, 1840
 ThirdPartyAccountLink 1841
 TimeSlot 1843
 TimeSlotHistory 1845
 Topic 1846
 TopicFeed 1850
 TopicLocalization 1856
 TwoFactorInfo 1861
 TwoFactorMethodsInfo 1862
 TwoFactorTempCode 1864

Objects (*continued*)

UndecidedEventRelation 1865
 User 1866
 UserAccountTeamMember 1907
 UserAppInfo 1910
 UserAppMenuCustomization 1911
 UserAppMenuCustomizationShare 1912
 UserAppMenuItem 1913
 UserConfigTransferButton 1917
 UserConfigTransferSkill 1917
 UserCustomBadge 1918
 UserCustomBadgeLocalization 1919
 UserFeed 1921
 UserLicense 1929
 UserLogin 1932
 UserMembershipSharingRule 1933
 UserPackageLicense 1935
 UserPreference 1936
 UserProfile 1937
 UserProfileFeed 1955
 UserProvAccount 1962
 UserProvAccountStaging 1965
 UserProvisioningConfig 1970
 UserProvisioningLog 1974
 UserProvisioningRequest 1975
 UserProvisioningRequestOwnerSharingRule 1981
 UserProvisioningRequestShare 1982
 UserProvMockTarget 1968
 UserRecordAccess 1984
 UserRole 1986
 UserServicePresence 1990
 UserShare 1991
 UserTeamMember 1993
 UserTerritory 1995
 UserTerritory2Association 1996
 VerificationHistory 1997
 vote 2003
 WebLink 2004
 WebLinkLocalization 2009
 WorkAccess 2014
 WorkAccessShare 2015
 WorkBadge 2016
 WorkBadgeDefinition 2019
 WorkBadgeDefinitionHistory 2022
 WorkBadgeDefinitionShare 2023
 WorkCoaching 2025
 WorkCoachingFeed 2026
 WorkCoachingHistory 2032
 WorkCoachingShare 2033

Objects (*continued*)

WorkFeedback 2035
 WorkFeedbackHistory 2036
 WorkFeedbackQuestion 2037
 WorkFeedbackQuestionHistory 2040
 WorkFeedbackQuestionSet 2041
 WorkFeedbackQuestionSetHistory 2042
 WorkFeedbackQuestionSetShare 2043
 WorkFeedbackQuestionShare 2044
 WorkFeedbackRequest 2046
 WorkFeedbackRequestFeed 2051
 WorkFeedbackRequestHistory 2056
 WorkFeedbackRequestShare 2057
 WorkFeedbackShare 2059
 WorkGoal 2060
 WorkGoalCollaborator 2066
 WorkGoalCollaboratorHistory 2067
 WorkGoalFeed 2069
 WorkGoalHistory 2075
 WorkGoalLink 2076
 WorkGoalShare 2077
 WorkOrder 2078
 WorkOrderFeed 2088
 WorkOrderHistory 2092
 WorkOrderLineItem 2093
 WorkOrderLineItemFeed 2101
 WorkOrderLineItemHistory 2105
 WorkOrderShare 2106
 WorkPerformanceCycle 2107
 WorkPerformanceCycleFeed 2110
 WorkPerformanceCycleHistory 2115
 WorkPerformanceCycleShare 2116
 WorkReward 2118
 WorkRewardFund 2120
 WorkRewardFundHistory 2122
 WorkRewardFundShare 2123
 WorkRewardFundType 2124
 WorkRewardFundTypeHistory 2127
 WorkRewardFundTypeShare 2128
 WorkRewardHistory 2130
 WorkRewardShare 2131
 WorkThanks 2132
 WorkThanksShare 2133
 WorkType 2134
 WorkTypeFeed 2136
 WorkTypeHistory 2139
 ObjectTerritory2AssignmentRule object 1301
 ObjectTerritory2AssignmentRuleItem object 1303
 ObjectTerritory2Association object 1304

- OpenActivity object 1306
 - OperatingHours object 1313
 - OperatingHoursFeed object 1315
 - OperatingHoursHistory object 1317
 - Opportunity object 1318
 - OpportunityCompetitor object 1329
 - OpportunityContactRole object 1330
 - OpportunityFeed object 1332
 - OpportunityFieldHistory object 1339
 - OpportunityHistory object 1341
 - OpportunityLineItem object 1343
 - OpportunityLineItemSchedule object 1349
 - OpportunityOverride object 1353
 - OpportunityOwnerSharingRule object 1356
 - OpportunityPartner object 1358
 - OpportunityShare object 1360
 - OpportunitySplit object 1362
 - OpportunitySplitType object 1364
 - OpportunityStage object 1367
 - OpportunityTag object 1369
 - OpportunityTeamMember object 1370
 - Order object 1373
 - OrderFeed object 1383
 - OrderHistory object 1389
 - OrderItem object 1391
 - OrderItemFeed object 1393
 - OrderItemHistory object 1400
 - OrderOwnerSharingRule object 1401
 - Organization object 1404
 - OrgWideEmailAddress object 1424
 - Outbound messaging
 - and notifications 2385
 - building a listener 2390
 - configuring up user profiles 2386
 - defining 2386
 - introduction 2383
 - notification calls in WSDL 2388
 - port restrictions 2386
 - security 2388
 - setting up 2385
 - tracking 2388
 - understanding 2384
 - viewing 2387
 - OwnedContentDocument object 1425
 - OwnerChangeOptions header 2369
- P**
- Package API access 88
 - PackageLicense object 1427
 - PackagePushError object 1429
 - PackagePushJob object 1431
 - PackagePushRequest object 1434
 - PackageSubscriber object 1436
 - PackageVersionHeader headers 2372
 - Partner object 1440
 - PartnerNetworkConnection object 1443
 - PartnerNetworkRecordConnection object 1446
 - PartnerNetworkSyncLog object 1449
 - PartnerRole object 1451
 - Performance cycle
 - data model 130
 - PerformQuickActionResult 2196
 - performQuickActions()
 - PerformQuickActionResult 2196
 - performQuickActions() call 2195
 - Period object 1453
 - Permission sets configuration 86
 - PermissionSet object 1455
 - PermissionSetAssignment object 1461
 - PermissionSetLicense object 1462
 - PermissionSetLicenseAssign object 1465
 - Persistent connections 2381
 - Picklists
 - multi-select picklists 2302
 - picklistValues 2302
 - restrictedPicklist 2302
 - single-select picklists 2302
 - PlatformAction object 1466
 - Port restrictions for outbound ports 90
 - Posts 981, 988
 - PresenceUserConfig object 1473
 - PresenceUserConfigProfile object 1476–1477
 - Pricebook2 object 1477
 - Pricebook2History object 1481
 - PricebookEntry object 1482
 - process() call 2196
 - ProcessDefinition object 1485
 - ProcessInstance object 1487
 - ProcessInstanceHistory object 1489
 - ProcessInstanceNode object 1495
 - ProcessInstanceState object 1492
 - ProcessInstanceWorkitem 1497
 - ProcessNode object 1499
 - Product2 object 1500
 - Product2Feed object 1506
 - ProductEntitlementTemplate object 1514
 - Profile object 1515
 - ProfileSkill object 1518

ProfileSkillEndorsement object 1520
ProfileSkillEndorsementFeed object 1521
ProfileSkillEndorsementHistory object 1525
ProfileSkillFeed object 1526
ProfileSkillHistory object 1532
ProfileSkillShare object 1533
ProfileSkillUser object 1534
ProfileSkillUserFeed object 1535
ProfileSkillUserHistory object 1540
PushTopic object 1541

Q

QuantityForecast object 1543
QuantityForecastHistory object 1551
query() call 2200
queryAll()
 limitations for external objects 2406
queryAll() call 2206
queryMore()

 limitations for external objects 2406
queryMore() call 2209
QueryOptions headers 2373
Question object 1556
QuestionDataCategorySelection 1560
QuestionReportAbuse object 1562
QuestionSubscription object 1563
QueueRoutingConfig object 1553
QueueSobject object 1565

Quickstart
 Generate the WSDL 6
 Import the WSDL 7
 Obtain a Developer Edition Organization 6
 Walk through code 8

QuickText object 1566
QuickTextHistory object 1568
QuickTextOwnerSharingRule object 1569
QuickTextShare object 1571
Quote object 1572
QuoteDocument object 1585
QuoteLineItem object 1587

R

Recent items list and update call 2368
RecordType object 1595
RecordTypeLocalization object 1598
Recycle Bin 2219
Relationships
 parent reference fields in child object 68
Reply object 1599

ReplyReportAbuse object 1602
Report object 1603
ReportFeed object 1607
ReportTag object 1614
ReputationLevel object 1616
ReputationLevelLocalization object 1617
ReputationPointsRule object 1619
Requirements for content type 2378
resetPassword() call 2338
ResourceAbsence object 1620
ResourceAbsenceFeed object 1624
ResourceAbsenceHistory object 1626
retrieve() call 2213
RevenueForecast object 1627
RevenueForecastHistory object 1636
Reward
 data model 129
Role, see UserRole 1986
RuleTerritory2Association object 1638

S

Salesforce instances 2378
Salesforce Sandbox 2377
SamISsoConfig object 1639
Sandbox 2377
Scontrol object 1644
ScontrolLocalization object 1647
SecureAgent object 1654
SecureAgentsCluster object 1655
Security
 and packages 88
 implicit restrictions for objects and fields 88
 outbound messaging 2388
 outbound port restrictions 90
 sharing 87
 user profile and permission sets configuration 86
Security token 86
SelfServiceUser object 1657
sendEmail() call 2334, 2340
sendEmailMessage call 2350
ServiceAppointment object 1660
ServiceAppointmentFeed object 1668
ServiceAppointmentHistory object 1670
ServiceAppointmentOwnerSharingRule object 1671
ServiceAppointmentShare object 1673
ServiceChannel object 1674
ServiceChannelStatus object 1676
ServiceContract object 1677
ServiceContractFeed object 1685

ServiceContractHistory object 1693
 ServiceContractOwnerSharingRule object 1694
 ServiceContractShare object 1696
 ServicePresenceStatus object 1698
 ServiceResource object 1699
 ServiceResourceCapacity object 1702
 ServiceResourceCapacityFeed object 1705
 ServiceResourceCapacityHistory object 1707
 ServiceResourceFeed object 1708
 ServiceResourceHistory object 1711
 ServiceResourceOwnerSharingRule object 1712
 ServiceResourceShare object 1714
 ServiceResourceSkill object 1715
 ServiceResourceSkillFeed object 1717
 ServiceResourceSkillHistory object 1719
 ServiceTerritory object 1720
 ServiceTerritoryFeed object 1725
 ServiceTerritoryHistory object 1727
 ServiceTerritoryMember object 1728
 ServiceTerritoryMemberFeed object 1733
 ServiceTerritoryMemberHistory object 1735
 Session ID expiration 84
 SessionHeader header 2374
 SessionPermSetAssignment object 1736
 setPassword() call 2353
 SetupAuditTrail object 1738
 SetupEntityAccess object 1740
 Sharing 87
 SignupRequest object 1742
 Site object 1749
 SiteDomain object 1755
 SiteHistory object 1756
 Skill object 1757
 SkillProfile object 1759
 SkillRequirement object 1759
 SkillRequirementFeed object 1761
 SkillRequirementHistory object 1764
 SkillUser object 1765
 SlaProcess object 1766
 SOAP headers 2355
 Solution object 1769
 SolutionFeed object 1773
 SolutionHistory object 1780
 SolutionStatus object 1782
 SolutionTag object 1783
 SOSDeployment object 1785
 SOSSession object 1787
 SOSSessionActivity object 1790
 SOSSessionHistory object 1791
 SOSSessionOwnerSharingRule object 1792
 SOSSessionShare object 1794
 SSL requirement 86
 Standard objects 131
 StaticResource object 1796
 StreamingChannel object 1798
 Stylesheets for Salesforce look and feel 95
 Support policy
 API 4
 Backward compatibility 4
 Deprecated API versions 4
 End-of-life notifications 4
 System fields 68
 SystemModstamp fields 68

T

TagDefinition object 1800
 Task object 1801
 TaskFeed object 1812
 TaskPriority object 1820
 TaskRelation object 1821
 TaskStatus object 1823
 TaskTag object 1825
 TaskWhoRelation object 1826
 TenantSecret object 1828
 Territory object 1830
 Territory2 object 1833
 Territory2Model object 1835
 Territory2ModelHistory object 1837
 Territory2Type object 1838
 TestSuiteMembership object 1000, 1840
 ThirdPartyAccountLink object 1841
 TimeSlot object 1843
 TimeSlotHistory object 1845
 TLS requirement 86
 Tooling API
 and the Enterprise and Partner WSDLs 73
 Topic object 1846
 TopicFeed object 1850
 TopicLocalization object 1856
 TwoFactorInfo object 1861
 TwoFactorMethodInfo object 1862
 TwoFactorTempCode object 1864
 Types of fields 32, 38, 40, 42

U

UndecidedEventRelation object 1865
 undelete() 2219
 Unicode 2381

Index

- update() call
 and recent items list [2368](#)
updateMru parameter in MruHeader header [2368](#)
upsert() call [2230](#)
User authentication [86](#)
User interface themes [95](#)
User object [1866](#)
User profile configuration [86](#)
User profiles
 configuring up for outbound messaging [2386](#)
UserAccountTeamMember object [1907](#)
UserApplInfo object [1910](#)
UserAppMenuCustomization object [1911](#)
UserAppMenuCustomizationShare object [1912](#)
UserAppMenuItem object [1913](#)
UserConfigTransferButton object [1917](#)
UserConfigTransferSkill object [1917](#)
UserCustomBadge object [1918](#)
UserCustomBadgeLocalization object [1919](#)
UserFeed object [1921](#)
UserLicense object [1929](#)
UserLogin object [1932](#)
UserMembershipSharingRule object [1933](#)
UserPackageLicense object [1935](#)
UserPreference object [1936](#)
UserProfile object [1937](#)
UserProfileFeed object [1955](#)
UserProvAccount object [1962](#)
UserProvAccountStaging object [1965](#)
UserProvisioningConfig object [1970](#)
UserProvisioningLog object [1974](#)
UserProvisioningRequest object [1975](#)
UserProvisioningRequestOwnerSharingRule object [1981](#)
UserProvisioningRequestShare object [1982](#)
UserProvMockTarget object [1968](#)
UserRecordAccess object [1984](#)
UserRole object [1986](#)
UserServicePresence object [1990](#)
UserShare object [1991](#)
UserTeamMember object [1993](#)
UserTerritory object [1995](#)
UserTerritory2Association object [1996](#)
UserTerritoryDeleteHeader header [2374](#)
UTF-8 [2381](#)
- V**
- VerificationHistory object [1997](#)
Version settings
 managed packages [81](#)
- Vote object [2003](#)
- W**
- WebLink object [2004](#)
WebLinkLocalization object [2009](#)
WorkAccess object [2014](#)
WorkAccessShare object [2015](#)
WorkBadge object [2016](#)
WorkBadgeDefinition object [2019](#)
WorkBadgeDefinitionHistory object [2022](#)
WorkBadgeDefinitionShare object [2023](#)
WorkCoaching object [2025](#)
WorkCoachingFeed object [2026](#)
WorkCoachingHistory object [2032](#)
WorkCoachingShare object [2033](#)
WorkFeedback object [2035](#)
WorkFeedbackHistory object [2036](#)
WorkFeedbackQuestion object [2037](#)
WorkFeedbackQuestionHistory object [2040](#)
WorkFeedbackQuestionSet object [2041](#)
WorkFeedbackQuestionSetHistory object [2042](#)
WorkFeedbackQuestionSetShare object [2043](#)
WorkFeedbackQuestionShare object [2044](#)
WorkFeedbackRequest object [2046](#)
WorkFeedbackRequestFeed object [2051](#)
WorkFeedbackRequestHistory object [2056](#)
WorkFeedbackRequestShare object [2057](#)
WorkFeedbackShare object [2059](#)
WorkGoal object [2060](#)
WorkGoalCollaborator object [2066](#)
WorkGoalCollaboratorHistory object [2067](#)
WorkGoalFeed object [2069](#)
WorkGoalHistory object [2075](#)
WorkGoalLink object [2076](#)
WorkGoalShare object [2077](#)
WorkOrder object [2078](#)
WorkOrderFeed object [2088](#)
WorkOrderHistory object [2092](#)
WorkOrderLineItem object [2093](#)
WorkOrderLineItemFeed object [2101](#)
WorkOrderLineItemHistory object [2105](#)
WorkOrderShare object [2106](#)
WorkPerformanceCycle object [2107](#)
WorkPerformanceCycleFeed object [2110](#)
WorkPerformanceCycleHistory object [2115](#)
WorkPerformanceCycleShare object [2116](#)
WorkReward object [2118](#)
WorkRewardFund object [2120](#)
WorkRewardFundHistory object [2122](#)

Index

- WorkRewardFundShare object [2123](#)
- WorkRewardFundType object [2124](#)
- WorkRewardFundTypeHistory object [2127](#)
- WorkRewardFundTypeShare object [2128](#)
- WorkRewardHistory object [2130](#)
- WorkRewardShare object [2131](#)
- WorkThanks object [2132](#)
- WorkThanksShare object [2133](#)
- WorkType object [2134](#)
- WorkTypeFeed object [2136](#)
- WorkTypeHistory object [2139](#)
- WSDL
 - version settings [81](#)

X

- XML compliance [2381](#)