



---

## Galaxy Database Dictionary

Version 7.8.6.0 (build 298)  
Jun 24, 2021

---

This document contains confidential and proprietary information of Gateway Ticketing Systems and may not be disclosed to any party without prior written consent of Gateway Ticketing Systems.

Copyright © 2002-2020 Gateway Ticketing Systems, Inc.  
All Rights Reserved.

## Table of Contents

Table of Contents	2
1 Galaxy Table Row Sizes	14
2 Introduction	15
2.1 Audit Columns	15
2.2 Data Type Conversion	15
3 Configuration	16
3.1 Table IDs	17
3.2 Activities	23
3.3 ActivityNotes	24
3.4 Agencies	25
3.5 AgencyControls	26
3.6 AgencyNodeRanges	27
3.7 AssociatedTicketRatios	28
3.8 BadChecks	29
3.9 BinDetails	30
3.10 BINs	31
3.11 BlockoutDates	32
3.12 BlockoutGroupDetails	33
3.13 BlockoutGroups	34
3.14 BonusPromotions	35
3.15 CancelReasons	36
3.16 Carriers	37
3.17 CCFTables	38
3.18 CCFNodes	39
3.19 CCFStatuses	40
3.20 ChangeFundOverride	41
3.21 ChangeFund	42
3.22 CITIES	43
3.23 COA	44
3.24 CodeTables	46
3.25 CodeTableValues	47
3.26 CodeValues	48
3.27 Companies	50
3.28 ConfigurationOptions	51
3.29 CountyCodes	70
3.30 Coupons	71
3.31 CreditCardUses	72
3.32 Currencies	73
3.33 CurrentLogons	74
3.34 DailyMenus	75
3.35 DateSpecificCapacityExcludedNodes	76
3.36 Denominations	77
3.37 Deposits	78
3.38 Disbursements	79
3.39 DisbursementDetails	80
3.40 DiscountPrivileges	81
3.41 Discounts	82
3.42 DiscountDetails	83
3.43 DiscountRequirements	84
3.44 DiscountValidations	85
3.45 DMACodes	86
3.46 DuplicateReferences	87
3.47 ExchangeDetails	88
3.48 ExchangeMethods	89
3.49 ExternalCalls	90
3.50 ExternalConnections	91
3.51 ExternalFieldConnections	92
3.52 ExternalSiteRecords	93

3.53 ExternalVariables	94
3.54 Fares	95
3.55 FieldAttributeGroups	96
3.56 FieldAttributes	97
3.57 FOPDetails	98
3.58 FOPMenuHeaders	99
3.59 FOPMenuDetails	100
3.60 FOPs	101
3.61 Holidays	103
3.62 HypercomBatchEntries	104
3.63 HypercomBatchResults	105
3.64 GalaxyMenus	106
3.65 GalaxyMenuItems	107
3.66 GeneratedStatements	108
3.67 GenericCalendars	109
3.68 GenericCalendarDetails	110
3.69 GroupSalesLimitHolds	111
3.70 GxItemGroups	112
3.71 GxItemGroupDetails	113
3.72 GxKeys	114
3.73 GxPrivateKeys	115
3.74 GxMasterKeys	116
3.75 IPAddresses	117
3.76 IPAddressGroups	118
3.77 ItemGroupReportHdrs	119
3.78 ItemGroupReportDtls	120
3.79 ItemImages	121
3.80 Items	122
3.81 ItemConnections	128
3.82 ItemPrices	129
3.83 ItemModifierGroupDetails	130
3.84 ItemStatements	131
3.85 ItemTranslations	132
3.86 ItemUpgrades	133
3.87 ItemVariations	134
3.88 Itineraries	135
3.89 ItineraryDetails	136
3.90 LeadMessages	137
3.91 LockerAssignments	138
3.92 LockerLog	139
3.93 Lockers	140
3.94 LockerTypes	141
3.95 LoyaltyAccountLog	142
3.96 LoyaltyAccounts	143
3.97 LoyaltyAccrualItems	144
3.98 LoyaltyProgramLevels	145
3.99 LoyaltyPrograms	146
3.100 LoyaltyRedemptionItems	147
3.101 MinMaxFares	148
3.102 Modifiers	149
3.103 ModifierGroups	150
3.104 ModifierGroupDetails	151
3.105 MultipleChoices	152
3.106 MultiSiteAccessCodeConfig	153
3.107 NameSuffixes	154
3.108 NameTitles	155
3.109 NodeDataConnections	156
3.110 NodeDataStatus	157
3.111 Nodes	158
3.112 NodeSerialNumbers	160
3.113 NodeVersionHistory	161

3.114 OnlineMenus	162
3.115 OrcaBlockedCards	163
3.116 OrcaCardPublicKey	164
3.117 OrcaResultTypes	165
3.118 Packages	166
3.119 PackageDetails	167
3.120 Pages	169
3.121 PaymentContractItems	170
3.122 PaymentContracts	171
3.123 PaymentContractStatuses	172
3.124 PaymentPlanConfig	173
3.125 PaymentPlanLog	174
3.126 PaymentPlanPriceData	175
3.127 PaymentPlanRecurrencePatterns	176
3.128 PaymentPlans	177
3.129 PkgInstances	178
3.130 PkgInstanceDetails	179
3.131 PlanConversions	180
3.132 PlanOps	181
3.133 PlannerFilters	182
3.134 Plugin Attributes	183
3.135 PluginConfigSections	184
3.136 Plugins	185
3.137 POSCreditTransactions	186
3.138 PostalCodeRangeDetails	187
3.139 PostalCodeRanges	188
3.140 PriceCalendars	189
3.141 PriceProgramGroups	190
3.142 PricePrograms	191
3.143 PriceProgramTimeRanges	192
3.144 PriceSchedules	193
3.145 ProfileControls	194
3.146 ProfilePrivs	195
3.147 Prompts	196
3.148 PromptSections	197
3.149 ProtocolIniSections	198
3.150 Protocols	199
3.151 PublishStatus	200
3.152 Reasons	201
3.153 RecurrencePatterns	202
3.154 Relationships	203
3.155 RelationshipTypes	204
3.156 RenewalOptions	205
3.157 ReportParameterDetails	206
3.158 ReportParameters	207
3.159 Reports	208
3.160 ReservationStatus	209
3.161 Rosters	210
3.162 RosterTemplates	211
3.163 RowChanges	212
3.164 SaleLimits	213
3.165 SaleLimitDetails	214
3.166 SettlementBatchDetails	215
3.167 SettlementBatches	216
3.168 SettlementConfig	217
3.169 Signatures	218
3.170 SiteItems	219
3.171 SiteMessages	220
3.172 Sites	221
3.173 Stocks	222
3.174 StockPackages	223

3.175 SurveyFields	224
3.176 SurveyReferences	225
3.177 Surveys	226
3.178 Taxes	227
3.179 TaxTableHeaders	228
3.180 TaxTableDetails	229
3.181 Templates	230
3.182 TemplateDetails	231
3.183 TerminalIDs	232
3.184 TerminalConfig	233
3.185 TktRanges	234
3.186 TktSets	235
3.187 TktSetDetails	236
3.188 TicketCodes	237
3.189 TicketNotes	238
3.190 TransactionalUpsellOptionRequirements	239
3.191 TransactionalUpsellOptions	240
3.192 TransportationTicketTypes	241
3.193 UpsellOptions	242
2 Emphasis Values	242
3.194 UserProfileItemGroups	243
3.195 UserProfilePassKindGroups	244
3.196 UserProfiles	245
3.197 UserSettings	246
3.198 ValueLinkConfig	247
3.199 VenueLimits	248
3.200 VenueLimitTotals	249
3.201 VisaNetTerminals	250
3.202 WaitLists	251
3.203 Warnings	252
3.204 WorkGroups	253
3.205 WorkGroupDetails	254
3.206 ZipCodes	255
<b>4 Access Control</b>	<b>256</b>
4.1 AccessCodeGroupDetails	257
4.2 AccessCodeGroups	258
4.3 AccessCodeOverrides	259
4.4 AccessCodePrintDetails	260
4.5 AccessCodes	261
4.6 ACPs	264
4.7 ACPGroups	265
4.8 ACPModes	266
4.9 ACPModeDetails	267
4.10 AcpStatus	268
4.11 ACSReports	269
4.12 ACSTotals	270
4.13 AdmissionHeaders	271
4.14 AdmissionDetails	272
4.15 Attendance	273
4.16 AttractionCounts	274
4.17 Attractions	275
4.18 AttributeDefinitions	276
4.19 AttributeCategories	277
4.20 AttributeDefinitionGroups	278
4.21 AttributeValues	279
4.22 BankHeaders	280
4.23 BankDetails	281
4.24 CalendarHeaders	283
4.25 CalendarDetails	284
4.26 Facility	286
4.27 FacilityAccess	287

4.28 FacilityCounts	288
4.29 FkeyTasks	289
4.30 HSAccessCodes	291
4.31 LockOuts	292
4.32 Media	293
4.33 MedialDDetails	295
4.34 MemberUsage	296
4.35 Operations	297
4.36 PendingUsage	298
4.37 Reentry	300
4.38 RFCS	301
4.39 RFCSUsage	302
4.40 RFCSOperations	304
4.41 TicketBanks	305
4.42 Tickets	306
4.43 SuperTickets	308
4.44 Usage	309
4.45 UsagePunches	317
4.46 ValidDates	318
<b>5 ACS Reservations</b>	<b>319</b>
5.1 ACSReservations	320
5.2 ACSReservationProfileConsequences	321
5.3 ACSReservationProfiles	322
5.4 ACSReservationRuleSetAdmissionHeaders	323
5.5 ACSReservationRuleSetConsequences	324
5.6 ACSReservationRuleSetNotifications	325
5.7 ACSReservationRuleSets	326
5.8 ACSReservationRuleSetStatuses	328
<b>6 Authentication</b>	<b>329</b>
6.1 AuthClaims	330
6.2 AuthClientGrantTypes	331
6.3 AuthClients	332
6.4 AuthClientScopes	333
6.5 AuthClientSecrets	334
6.6 AuthIdentityClaims	335
6.7 AuthIdentityResources	336
6.8 AuthPersistedGrants	337
6.9 AuthResources	338
6.10 AuthScopes	339
<b>7 Blacklist</b>	<b>340</b>
7.1 Blacklists	341
7.2 BlacklistMembers	342
7.3 BlacklistActions	343
7.4 BlacklistActionTypeNames	344
7.5 BlacklistDefaultActions	345
<b>8 Debit</b>	<b>346</b>
8.1 DebitTypes	347
8.2 DebitCards	348
8.3 DebitCardLog	349
8.4 DebitLockouts	350
8.5 SVDataConnectionValues	351
<b>9 eGalaxy</b>	<b>352</b>
9.1 AppendedOrderLines	353
9.2 CustomerGroups	354
9.3 CustomerGroupDetails	355
9.4 eGalaxyConfig	356
9.5 eGalaxyErrorLog	357
9.6 eGalaxyErrorSeverities	358
9.7 eGalaxyJournalFiscalSummaryQueue	359
9.8 eGalaxyPasswordResetTokens	360

9.9 eGalaxyQueryTicketFieldRestrictions	361
9.10 eGalaxySessions	365
9.11 eGalaxySourceConfigOptions	366
9.12 eGalaxySourceIPAddresses	367
9.13 eGalaxySourceMediaDefinitions	368
9.14 eGalaxySources	369
9.15 eGalaxyTransactionIDs	371
9.16 EmailStatistics	372
9.17 EmailStatisticDetails	373
9.18 EmailStatisticSubDetails	374
9.19 ErrorNotificationOptions	375
9.20 GxProcessTriggers	376
9.21 HSConfig	377
9.22 OnlineExchangeHistory	378
9.23 RejectedOrderCommandErrors	379
9.24 SequenceNumbers	380
<b>10 Event Store</b>	<b>381</b>
10.1 GxEventStore	382
10.2 GxEventStoreConsumerDetails	384
10.3 GxEventStoreConsumers	385
10.4 GxEventStoreQueue	386
10.5 GxEventStoreTags	387
<b>11 Gift</b>	<b>388</b>
11.1 Appeals	389
11.2 AppealSolicitationConnections	390
11.3 Campaigns	391
11.4 CampaignAppealConnections	392
11.5 Funds	393
11.6 GiftAidCategoryPriceData	394
11.7 GiftAidClaims	395
11.8 Gifts	396
11.9 GiftDetails	397
11.10 GiftFieldsResponses	398
11.11 Solicitations	399
<b>12 Journal</b>	<b>400</b>
12.1 JnlHeaders	401
12.2 JnlDetails	404
12.3 JnlAssignSeats	407
12.4 JnlBonuses	408
12.5 JnlCapacities	409
12.6 JnlCashoutShifts	410
12.7 JnlCCFTables	411
12.8 JnlCCFMemos	412
12.9 JnlCCFNodes	413
12.10 JnlCCFStatuses	414
12.11 JnlChangeFunds	415
12.12 JnlContacts	416
12.13 JnlCurrencies	417
12.14 JnlDCCDrafts	418
12.15 JnlDebits	419
12.16 JnlDenominations	420
12.17 JnlDeposits	421
12.18 JnlDigitalSignatures	422
12.19 JnlDiscounts	423
12.20 JnlDrafts	424
12.21 JnleSelectPlusDrafts	426
12.22 JnlExpressBusbills	427
12.23 JnlExpressSurcharges	428
12.24 JnlFiscalSummaryDetails	429
12.25 JnlFiscalSummaryHeaders	430

12.26 JnlFiscalSummaryTaxDetails	431
12.27 JnlFiscalTransactionArchiving	432
12.28 JnlGifts	433
12.29 JnlHostDataFields	434
12.30 JnlItemQuantityReductions	436
12.31 JnlItems	437
12.32 JnlItineraries	439
12.33 JnlLoyaltyAccounts	440
12.34 JnlModifiers	441
12.35 JnlNoSales	442
12.36 JnlOverShorts	443
12.37 JnlPackages	444
12.38 JnlPaidIOs	445
12.39 JnlPayments	446
12.40 JNLPTOs	447
12.41 JnlReasons	448
12.42 JnlReceipts	449
12.43 JnlRecharges	450
12.44 JnlReissues	451
12.45 JnlRenewalCredits	452
12.46 JnlReprints	453
12.47 JnlResellerDetails	454
12.48 JnlReservations	455
12.49 JnlStocks	456
12.50 JnlSuperTickets	457
12.51 JnlSurcharges	458
12.52 JnlSurveys	459
12.53 JnlSVLookups	460
12.54 JnlTaxes	461
12.55 JnlTickets	462
12.56 JnlTranRefunds	466
12.57 JnlTripSummaries	467
12.58 JnlUDFFields	468
12.59 JnlUpgrades	469
12.60 JnlUpsells	470
12.61 JnlUsage	471
12.62 JnlVoids	472
12.63 JnlVouchers	473
12.64 JournalBreakupLog	474
12.65 MemoTbl	475
12.66 SequenceNumbers	477
12.67 Settlements	478
12.68 ShiftStatus	479
<b>13 Kiosk / Web Store / Sales Channel</b>	<b>480</b>
13.1 CategoryPriceData	481
13.2 MarketingMessages	482
13.3 Merchants	483
13.4 PassBenefits	484
13.5 Promotions	485
13.6 PromotionCodes	486
13.7 PromotionOffers	487
13.8 PromotionValidations	488
13.9 ResellerPriceSchedules	488
13.10 ResellerSalesChannelDetails	489
13.11 SalesChannelOptions	490
13.12 SalesChannelPublishCommands	491
13.13 SalesChannelPublishCommandDetails	492
13.14 SalesChannels	493
13.15 SalesChannelDeliveryMethods	494
13.16 SalesChannelDetails	495
13.17 SalesChannelIPAddresses	499

13.18 SalesChannelLanguages	500
13.19 SalesChannelLoyaltyPoints	501
13.20 SalesChannelNodes	502
13.21 SalesChannelPublishLog	503
13.22 SystemLogons	504
13.23 TranslationEventNames	505
13.24 TranslationFields	506
13.25 TranslationLanguages	507
13.26 TranslationTables	508
13.27 TranslationValues	509
13.28 Upsell	510
13.29 WSHTML	511
13.30 WSHtmlConnections	513
13.31 WebTemplates	514
<b>14 Manager Workstation</b>	<b>515</b>
14.1 GXUserKeywords	516
14.2 GxUsers	517
14.3 Messages	518
14.4 MWSAlerts	519
14.5 MWSOperations	520
14.6 PasswordHistory	521
<b>15 Order Entry</b>	<b>522</b>
15.1 ActiveCustCategories	523
15.2 Addresses	524
15.3 ARAccounts	525
15.4 ARCreditMemoLines	526
15.5 ARCreditMemoPayments	528
15.6 ARCreditMemos	529
15.7 ARInvoiceLines	530
15.8 ARInvoiceOrders	532
15.9 ARInvoicePayments	533
15.10 ARInvoices	534
15.11 ARTerms	535
15.12 CommunicationMethods	536
15.13 ContactConnections	537
15.14 ContactDonorTypes	538
15.15 ContactLists	539
15.16 ContactListDetails	540
15.17 ContactPaymentInfo	541
15.18 CustCategories	542
15.19 CustCategoryTotals	543
15.20 CustContacts	544
15.21 CustContactsConsent	546
15.22 CustContactsConsentProcess	547
15.23 Customers	548
15.24 DeliveryMethods	550
15.25 DeliveryMethodGroups	552
15.26 DeliveryMethodGroupDetails	553
15.27 FilterDefinitions	554
15.28 FilterConnections	555
15.29 FOPSets	556
15.30 FOPSetDetails	557
15.31 GroupSalesCodes	558
15.32 GXUserFields	559
15.33 ItemConstraints	560
15.34 ItemDeliveryConversions	561
15.35 ItemGroupDetails	562
15.36 ItemGroups	563
15.37 ItemRates	564
15.38 OEBatchPrintDetails	565

15.39 OEBatchPrintHeader	566
15.40 OECreditAccountUpdateLog	567
15.41 OEGroupVisits	568
15.42 OEGuests	569
15.43 OEMarkedOrders	570
15.44 OESTatementGroups	571
15.45 OESuspendedOrders	572
15.46 OEUserFields	573
15.47 OEWTemplateGroups	574
15.48 OrderCommands	575
15.49 OrderCommunications	576
15.50 OrderDetails	577
15.51 OrderDiscounts	579
15.52 OrderDrafts	580
15.53 OrderItemDetails	581
15.54 OrderLineDiscounts	582
15.55 OrderLineDiscountReplacements	583
15.56 OrderLines	584
15.57 OrderLineGiftDetails	587
15.58 OrderLineJointMembers	588
15.59 OrderLineSeats	589
15.60 OrderLineTaxAmounts	590
15.61 OrderLineTransportationTickets	591
15.62 OrderLineUpgrades	592
15.63 OrderLineUpsells	593
15.64 OrderPayments	594
15.65 OrderRateTables	595
15.66 OrderReprintHistory	596
15.67 OrderRules	597
15.68 Orders	598
15.69 OrderSurveys	601
15.70 OrderTaxDetails	602
15.71 OrderTransactions	603
15.72 PhoneNumbers	604
15.73 SalesPrograms	605
15.74 SalesProgramTypes	606
15.75 SerializedStatements	607
15.76 SerializedStatementDetails	608
15.77 SMSTemplates	609
15.78 SPCConnections	610
15.79 SPPriceCalendars	611
15.80 SPPriceSchedules	612
15.81 Statements	613
15.82 TaxSetDetails	614
15.83 TaxSets	615
15.84 TempSpendingHistory	616
15.85 UserFieldLabels	617
<b>16 Passes</b>	<b>618</b>
16.1 JointMembers	619
16.2 PassEditFields	620
16.3 PassHistory	622
16.4 PassHistoryRecordLog	623
16.5 PassPkgInstances	624
16.6 Passes	625
16.7 PassKinds	628
16.8 PassKindGroups	631
16.9 PassKindGroupDetails	632
16.10 PassKindsRE	633
16.11 PassReqLog	634
16.12 PassRestrictions	635

<b>17 Photo</b>	<b>636</b>
17.1 Pictures	637
<b>18 Resource Management</b>	<b>638</b>
18.1 EventManagementActivity	639
18.2 EventManagementSeatActivity	640
18.3 EventManagementSessions	641
18.4 EventSaleLimits	642
18.5 HoldDetails	643
18.6 HoldTemplates	644
18.7 RentalSerial	645
18.8 RMAudioTypes	646
18.9 RMCapacity	647
18.10 RMCapacityHold	648
18.11 RMCommissions	649
18.12 RMCommissionTypes	650
18.13 RMDistributors	651
18.14 RMEventPrograms	652
18.15 RMEventProgramDetails	653
18.16 RMEvents	654
18.17 RMEventTypes	655
18.18 RMFeatures	656
18.19 RMFeatureTypes	657
18.20 RMFeatureRatings	658
18.21 RMReservationHold	659
18.22 RMReservations	660
18.23 RMResources	661
18.24 RMResourceTypes	662
18.25 RMShows	663
18.26 RSEventPrices	664
18.27 RSEventPriceDetails	665
18.28 RSEventSeatHolds	666
18.29 RSEventSeatMapObjects	667
18.30 RSEventSeatMaps	668
18.31 RSEventSeats	669
18.32 RSEventSections	670
18.33 RSHoldCodes	671
18.34 RSHoldCodeUserProfiles	672
18.35 RSPrices	673
18.36 RSPriceDetails	674
18.37 RSResourceMaps	675
18.38 RSRowPrices	676
18.39 RSRows	677
18.40 RSSeatHolds	678
18.41 RSSeatMapObjects	679
18.42 RSSeatMapPriceConnections	680
18.43 RSSeatMaps	681
18.44 RSSeatPrices	682
18.45 RSSeatMapPrices	683
18.46 RSSeats	684
18.47 RSSeatDisplayStatuses	685
18.48 RSSectionMaps	686
18.49 RSSectionPrices	687
18.50 RSSections	688
18.51 RSTests	689
18.52 RSTestGroups	690
18.53 RSTestGroupDetails	691
18.54 RSTestTransactions	692
18.55 SeatAssignmentImports	693
<b>19 SIAE</b>	<b>694</b>
19.1 JnlSIAE	695

19.2 SIAEAbsoluteItemCounters	697
19.3 SIAECalendarDetails	698
19.4 SIAECalendarHeaders	699
19.5 SIAEEventOrganizers	700
19.6 SIAEEvents	701
19.7 SIAEItems	702
19.8 SIAEOptions	703
19.9 SIAEReductionCodes	704
19.10 SIAEVenueDetails	705
19.11 SIAEVenues	706
19.12 SIAEVenueUsage	707
19.13 SIAEVirtualTickets	708
19.14 SIAEVoidReasons	709
19.15 SIAEXMLReportTrack	710
<b>20 Smart Upsell</b>	<b>711</b>
20.1 SmartUpsellSalesGoals	712
20.2 SmartUpsellSalesGoalDetails	713
20.3 SmartUpsellRules	714
<b>21 Voucher</b>	<b>715</b>
21.1 MarkedVouchers	716
21.2 Vouchers	717
21.3 VoucherDetails	718
21.4 VoucherBatches	719
21.5 VoucherMarketing	720
<b>22 Web Store</b>	<b>721</b>
22.1 CustContactsLogin	722
22.2 EditedBillingInfo	723
22.3 OrderCreditCardLog	724
22.4 RejectedOrderErrors	725
22.5 RejectedSurveyResultErrors	726
22.6 RetrievePrices	727
22.7 Sched1	728
22.8 SessionAssign	729
22.9 SurveyResults	730
22.10 SurveyResultDetails	731
22.11 WebFares	732
22.12 WebScheds	733
22.13 WebSchedDetail	734
22.14 WebStoreErrors	735
22.15 WSActivityLog	736
22.16 WSConfig	737
22.17 WSConfigDetails	739
22.18 WSConfigHeaders	740
22.19 WSKeyValue	741
22.20 WSLocalization	742
22.21 WSLocalizationGroups	743
22.22 WSLocalizationGroupModules	744
22.23 WSLog	745
22.24 WSLogConfig	746
22.25 WSOrderPaymentsLog	747
22.26 WSPaymentBootstrapDetails	748
22.27 WSPaymentBootstrapHeaders	749
22.28 WSPaymentConfig	750
22.29 WSPaymentConfigDetails	751
22.30 WSPaymentConfigHeaderOverrides	752
22.31 WSPaymentConfigHeaders	753
22.32 WSPaymentThemeDetails	754
22.33 WSPugins	755
22.34 WSPublishingLog	756
22.35 WSPublishStatus	757

22.36 WSSecurityIPAddresses	758
<b>23 Miscellaneous</b>	<b>759</b>
23.1 ApiUsage	760
23.2 Countries	761
23.3 GatewayCounters	762
23.4 GatewayLocks	763
23.5 GXEventLog	764
23.6 GXExchangeRates	768
23.7 GXFileConnections	769
23.8 GXFiles	770
23.9 GXProjections	771
23.10 GxRecordLog	772
23.11 GxTriggers	773
23.12 LogEntries	775
23.13 MessageQueue	776
23.14 MessageQueueFiles	777
23.15 Notes	778
23.16 REConfig	779
23.17 RFIDMaps	781
23.18 SecurityAnswers	782
23.19 SecurityLog	783
23.20 StateRegions	784
23.21 TicketLog	785
<b>24 Auto Upgrade Database</b>	<b>786</b>
24.1 ManualUpgradeTables	787
24.2 TableIndexDefs	788
24.3 UpgradeDefaultValues	789
24.4 UpgradeHistory	790
24.5 UpgradeLogs	791
24.6 UpgradeSettings	792
<b>25 External Application</b>	<b>793</b>
25.1 BatchTicketImports	794
25.2 COAReportExport	795
25.3 OiRejectedFileNames	796
25.4 RejectedOrderHeader	797
25.5 RejectedOrderDetails	798
25.6 UserDiscrepancyLog	799
<b>26 Utilities</b>	<b>800</b>
26.1 GTS_SP_CopyDatabase	801
26.2 GTS_SP_PurgeGXRecordLog	802
26.3 GTS_SP_PurgeRowChanges	803

## 1 Galaxy Table Row Sizes

Execute the following SQL query to gather the table row sizes for all available tables in the current database:

```
SELECT a.Name, sum(b.length)
FROM sysobjects a, syscolumns b, systypes c
WHERE a.id = b.id
    AND b.xtype = c	xtype
    AND c.name <> 'boolean'
    AND a.type = 'U'
GROUP BY a.Name
ORDER BY a.Name
```

## 2 Introduction

This document serves as the Database Dictionary for Galaxy.

### 2.1 Audit Columns

With the exception of auxiliary tables for the journal, all tables contain standard audit columns. To save space in this document, these columns are not shown, but should always be assumed to be present on every table.

Column	Type	Allow Nulls	Description
RecordVersion	Int	N	A number that acts as the revision number of this record. Newly inserted records have a RecordVersion1 and it is incremented every time the record is updated.
LastUpdate	DateTime	N	Timestamp provided by the server to indicate when the record was last updated.
LastUpdatedBy	VarChar(20)	Y	The server-provided username of the person who last updated the record or it has a pattern of "2-letter application abbreviation + 5-digit node no + "," + 3-digit Galaxy UserID"

### 2.2 Data Type Conversion

The table listed below maps the data types used in this document to the specified database server.

Documented Type	MS-SQL	Oracle
Bit	Bit	Char(2)
Char	Char	Char
DateTime	DateTime	Date
Varchar	Varchar	Varchar2
Float	Float	Number(*)
Image	Image	LongRaw
Int	Int	Number(10, *)
Money	Money	Number(*)
Text	Text	Long

## 3 Configuration

### 3.1 Table IDs

The Chart below lists the IDs for Btree and DBMS tables.

TableID Values

Value	Gateway Constant Name	Description
1	COA_DBID	Chart of Accounts
2	AGENCIES_DBID	Agencies
3	STOCK_DBID	Ticket Stock
4	PRODUCT_DBID	Product
5	RATES_DBID	Rates
6	DISBURSE_DBID	Disbursements
7	CLASSES_DBID	Agent Classes (User Profile)
8	ITEM_DBID	Items
9	CUSTCAT_DBID	Customer Categories
10	CUSTDISC_DBID	Customer Discounts
11	CUSTRATE_DBID	Customer Rates
12	CUSTOMER_DBID	Customers
13	DISCOUNT_DBID	Discounts
14	PASS_DBID	SQL Passes Table
15	PASSKIND_DBID	Pass Kinds
16	DMA_DBID	DMA Code (demographic market area)
17	ZIP_DBID	Zip Codes
18	COUNTY_DBID	Counties
19	PROMPTS_DBID	Formatted Prompts
20	BIN_DBID	BankCard Formats
21	HQTASKS_DBID	Concentrator Tasks
22	JNLINDEX_DBID	Journal Index
23	PICTURES_DBID	Pictures (Images)
24	ONLMENU_DBID	Online Menu
25	ORDOPEN_DBID	Orders - Open
26	ORDERDTL_DBID	Order Details
27	ORDERMEM_DBID	Order
28	EVENT_DBID	SQL RMEvents Table
29	EVENTTYP_DBID	Event Types
30	CONSIGN_DBID	Consignments
31	INVHDR_DBID	Invoice Headers
32	INVDTL_DBID	Invoice Details
33	ORDERSTM_DBID	Order Statements
34	PRINTDEF_DBID	Printer Definitions
35	ACCESS_DBID	SQL Access Codes Table
36	MEDIA_DBID	Media Definitions
37	TKTRANGE_DBID	Ticket Ranges
38	USAGELOG_DBID	Usage Log
39	FACILITY_DBID	Facilities
40	POSITIVE_DBID	
41	ACP_DBID	Access Control Points
42	VALID_DBID	
43	REENTRY_DBID	ReEntries
44	LOCKOUT_DBID	SQL Lockouts Table
45	ACSREPDB_DBID	User Defined Access Reports
46	ACSTOTAL_DBID	Access Code Totals
47	FKEYTASK_DBID	FKey Tasks
48	ACPSTAT_DBID	
49	PAGEMENU_DBID	Lead Paging Messages
50	TKTSALES_DBID	Ticket Sales
51	LIFTBAD_DBID	
52	LIFT_DBID	
53	HOTEL_DBID	
54	INVMEM_DBID	
55	ORDRCLSD_DBID	Orders - closed
56	AGENT_DBID	Agents (users)
57	CAPACITY_DBID	Resource Capacity
58	SECTION_DBID	Event Sections
59	ADDRESS_DBID	Addresses
60	SURVKIND_DBID	Survey Kinds
61	SURVANS_DBID	Survey Answers
62	SURVDESC_DBID	Survey Descriptions
63	EVSHOW_DBID	Event Shows

64	COUPONS_DBID	Coupons
65	TKTSETS_DBID	Ticket Sets
66	SHOW_DBID	Shows
67	PASSRSTR_DBID	Pass Kind Restrictions
68	PASSMEMO_DBID	
69	PASSEDIT_DBID	
70	EXCHANGE_DBID	Exchange rate is not db; value used by ccf
71	FOP_DBID	Forms of Payment
72	CONFIG_DBID	Configuration
73	GALAXY_MENU_DBID	Galaxy Menu
74	GALAXY_MENUITEM_DBID	Galaxy Menu Items
75	SALELOCK_DBID	Blockout Dates
76	COMPANY_DBID	Companies
77	DIALER_DBID	Phone Dialer
78	PROTOCOL_DBID	Credit Card Protocols
79	MESSAGE_DBID	Messages
80	NODES_DBID	Nodes
81	TKTCODES_DBID	Ticket Codes
82	BLKTEGRPS_DBID	Blockout Groups
83	BLKTDTLS_DBID	Blockout Group Details
84	HOLIDAYS_DBID	Holidays
85	CALENDAR_DBID	Calendars
86	CALDTLS_DBID	Calendar Details
87	BANKS_DBID	Banks
88	BANKDTLS_DBID	Bank Details
89	OPER_DBID	Operations
90	ADMIT_DBID	Admission Lists
91	ADMITDTL_DBID	Admission List Details
92	EXCH_DBID	Exchange Methods
93	EXCHDTLS_DBID	Exchange Details
94	ITEM_GROUP_DBID	Item Groups
95	ITEM_GROUP_DETAILS_DBID	Item Group Details
96	USER_PROFILE_ITEM_GROUPS_DBID	User Profile Item Groups
97	AGENCY_CONTROL_DBID	Agency Controls
98	ITEM_GROUP_REPORTS_HDR_DBID	Item Group Reports
99	ITEM_GROUP_REPORTS_DTL_DBID	Item Group Report Details
100	DENOM_DBID	Denominations
101	HNDSHK_DBID	Handshake
102	ACP_MODE_DBID	ACP Mode
103	ACP_MODE_DTL_DBID	ACP Mode Details
104	EXTERNAL_CALLS_DBID	External Calls
105	RECONFIG_DBID	Raiser's Edge Configuration
106	UPGRADE_DBID	Upgrades
107	DBMS_CUSTOMERS_DBID	SQL Customers Table
108	DBMS_ORDERS_DBID	SQL OrderLines Table
109	DBMS_ORDERLINES_DBID	SQL OrderLines Table
110	DBMS_ADDRESSES_DBID	SQL Addresses Table
111	DBMS_ACCOUNTS_DBID	SQL ARAccounts Table
112	DBMS_NOTES_DBID	SQL Notes Table
113	DBMS_CONTACTS_DBID	SQL CustContacts Table
114	DBMS_ORDER_TRANSACTIONS_DBID	SQL OrderTransactions Table
115	DBMS_ORDER_PAYMENTS_DBID	SQL OrderPayments Table
116	DBMS_ORDER_DRAFTS_DBID	SQL OrderDrafts Table
117	DBMS_ORDER_DETAILS_DBID	SQL OrderDetails Table
118	DBMS_ORDER_ITEM_DETAILS_DBID	SQL OrderItemDetails Table
119	DBMS_ORDER_TAX_DETAILS_DBID	SQL OrderTaxDetails Table
120	DBMS_CONTACTCONNECTIONS_DBID	SQL ContactConnections Table
121	DBMS_SPCONNECTIONS_DBID	SQL SPConnections Table
122	DBMS_ARINVOICES_DBID	SQL ARInvoices Table
123	DBMS_ARINVOICELINES_DBID	SQL ARInvoiceLines Table
124	DBMS_ARCREDITMEMOS_DBID	SQL ARCreditMemos Table
125	DBMS_ARCREDITMEMOLINES_DBID	SQL ARCreditMemoLines Table
126	DBMS_OEBATCHPRINTDETAILS_DBID	SQL OEBatchPrintDetails Table
127	DBMS_OEGROUPVISITS_DBID	SQL OEGroupVisits Table
128	DBMS_OEMARKEDORDERS_DBID	SQL OEMarkedOrders Table
129	DBMS_OEUSERFIELDS_DBID	SQL OEUserFields Table
130	DBMS_ORDERCOMMANDS_DBID	SQL OrderCommands Table
131	DBMS_OEBATCHPRINTHEADER_DBID	SQL OEBatchPrintHeader Table
132	DBMS_ORDER_LINE_TAX_AMOUNTS_DBID	SQL OrderLineTaxAmounts Table

133	TAX_TABLE_HDR_DBID	Tax Table Headers
134	TAX_TABLE_DTL_DBID	Tax Table Details
135	DBMS_CONTACT_LISTS_DBID	SQL ContactLists Table
136	DBMS_CONTACT_LIST_DETAILS_DBID	SQL ContactListDetails Table
137	DBMS_OEGUESTS_DBID	SQL OEGuests Table
138	BONUS_PROMOTIONS_DBID	Bonus Promotions
139	DEF_DAILY_DBID	Default Daily Menus
140	GX_KEY_DBID	Encryption Keys
141	DBMS_RELATIONSHIP_TYPES_DBID	SQL Relationship table
142	DBMS_RELATIONSHIPS_DBID	SQL Relationship Types table
143	DBMS_NAME_TITLES_DBID	SQL Name Titles table
144	DBMS_NAME_SUFFIXES_DBID	SQL Name Suffixes table
145	DBMS_CAMPAIGNS_DBID	SQL Campaigns table
146	DBMS_FUNDS_DBID	SQL Funds table
147	DBMS_APPEALS_DBID	SQL Appeals table
148	DBMS_SOLICITATIONS_DBID	SQL Solicitations table
149	DBMS_GIFTS_DBID	SQL Gifts table
150	DBMS_GIFT_DETAILS_DBID	SQL Gift Details table
151	DBMS_PASS_HISTORY_DBID	SQL Pass History table
152	DBMS_EXTERNAL_FIELD_CONNECTIONS_DBID	SQL External Fields Connections table
153	DBMS_USAGE_DBID	SQL Usage table
154	DBMS_GX_USER_FIELDS_DBID	SQL Gx User Fields table
155	DBMS_CAMPAIGN_APPEAL_CONNECTIONS_DBID	SQL Campaign Appeal Connections table
156	DBMS_APPEAL_SOLICITATION_CONNECTIONS_DBID	SQL Appeal Solicitation Connections table
157	DBMS_TICKETS_DBID	SQL Tickets table
158	ACCESS_CODE_GROUP_DBID	Access Code Groups
159	ACCESS_CODE_GROUP_DETAIL_DBID	Access Code Group Details
160	RFCS_DBID	RFCS Items
161	RFCS_OPERATION_DBID	RFCS Operations
162	PRICE_PROGRAM_DBID	Price Programs
163	PRICE_PROGRAM_TIME_RANGE_DBID	Price Program Time Range
164	PRICE_SCHEDULE_DBID	Price Schedule
165	PRICE_CALENDAR_DBID	Price Calendar
166	DBMS_SUPER_TICKETS_DBID	SQL Super Tickets table
167	PACKAGE_DBID	Packages
168	PKG_DTL_DBID	Package Details
169	DBMS_SCHED1_DBID	
170	DISBURSE_DTL_DBID	Disbursements
171	DISCOUNT_DTL_DBID	Discount Details
172	DISCOUNT_REQ_DBID	Discount Requirements
173	MEDIA_DETAILS_DBID	Media Details
174	SURVEY_FIELDS_DBID	Survey Fields
175	SURVEY_REF_DBID	Survey Reference
176	CARRIER_DBID	Companies providing intercity or other scheduled service suitable for using the Quasar routing and scheduling engine.
177	CARRIER_EXPRESS_DBID	Defines parcel express (shipping) parameters for each carrier
178	CARRIER_HOLIDAY_DBID	The list of official holidays for each carrier
179	CARRIER_PTO_DBID	Defines the Galaxy item, fee and cash advance limits for a prepaid ticket order for each carrier/agency
180	CARRIER_REFUND_DBID	Defines refund fee charged by the specified carrier
181	CITY_BULLETIN_DBID	Text information for specific cities that can be sent to POS stations to alert the ticket agent of special conditions
182	CITY_INFO_DBID	Hours, special services, etc in free form text
183	COMMISSION_RATE_DBID	Commission rate in Chart of Accounts varies by agencies in TX
184	CONNECT_TIME_DBID	The minimum time required to allow a connection between two carriers at a specific location (city).
185	FARE_DBID	Defines fares for specific city pairs including ticket types and restrictions such as advance purchase or day-of-week.
186	FARE_CLASS_DBID	Table to define fare classes and how they can be combined
187	HONOR_CARRIER_DBID	Carriers that can issue and/or honor a fare - this table is detail to the Tariffs table
188	LATLON_DBID	Created from LatLonTasa.txt and used to populate Lat/Lon fields in Cities when importing cities from TX data
189	MANUAL_UPGRADE_DBID	
190	MRT_DBID	Mileage Rate Tables - the cost for a range of miles expressed as cents per mile or fixe price
191	MRT_DETAIL_DBID	The range of miles and cost for each mileage rate table
192	Q1ROUTE_DBID	Stored data-specific routes that have been previously quoted on a machine. Used to bypass route gen after the first time a city pair has been quoted at the POS
193	RESTRICTION_DBID	Restriction codes that be be assigned to a fare record
194	SEGMENT_MILE_DBID	Temporary Table of schedule and tariff miles for each unique schedule segment - used to construct a spreadsheet for each carrier to enter tariff miles
195	SURCHARGE_CITY_TARIFF_DBID	Tariffs by City
196	SURCHARGE_CITY_DBID	Surcharges by City
197	SURCHARGE_ZONE_DBID	Surcharges by Zone
198	SURCHARGE_ZONE_TARIFF_DBID	Surcharge Zone Tariffs
199	TARIFF_DBID	General description of each tariff - see HonoringCarriers for tariff detail

201	TARIFF_MILE_DBID	Agreed miles to calculate mileage-based fares
202	TEMPLATE_DBID	Ticket type templates to assign discounted fares to function keys. This is the primary link to the rest of Galaxy because each ticket type includes the PLU item number for the Galaxy Items table.
203	TEMPLATE_DETAIL_DBID	Up to 20 records representing F1 - F10 for one way and round trip tickets
204	TX_ROUTE_DBID	Routes derived from TX products - used primarily to compare TX and Quasar quotes
205	ZB_FILTER_DBID	Sero Break Filter
206	ZB_SCHEDULE_DBID	Zero Break Schedules
207	ZB_TABLE_DBID	Zero Break Table - city pairs with direct service; i.e., "Destin" occurs on the same schedule as "Origin" and its segment number is greater than Origin's segment number
208	ZONE_DBID	A list of named zones
209	ZONE_DETAIL_DBID	The list of cities assigned to a zone
210	ZONE_FARE_DBID	Zone-based fares, primarily used to implement Greyhound Primary and Mini Market fares
211	MERCHANT_DBID	Merchants
212	SALES_CHANNEL_DETAIL_DBID	Sales Channel Details
213	WSHTML_DBID	Web Store HTML
214	CITY_DBID	List of Cities
215	STATE_DBID	List of States
216	TENANT_DBID	List of Tenants
217	DEBIT_TYPE_DBID	
218	DBMS_CONTACT_PAYMENT_INFO_DBID	
219	DBMS_WAIT_LISTS_DBID	
220	DBMS_CUSTOMER_CATEGORIES_DBID	
221	DISCOUNT_VALIDATIONS_DBID	Discount Validations
222	ITEM_CONSTRAINT_DBID	Item Constraints
223	ITEM_RATES_DBID	Item Rates
224	MULTIPLE_CHOICE_DBID	Multiple Choices
225	SALES_PROGRAM_DBID	Sales Programs
226	GX_ITEM_GROUP_DBID	Galaxy Item Groups
227	DELIVERY_METHOD_GROUP_DBID	Delivery Method Groups
228	DELIVERY_METHOD_DBID	Delivery Method Group Details
229	DBMS_PASSKIND_DBID	
230	DBMS_PASSEDIT_DBID	
231	PROMOTION_DBID	Promotions
232	PROMOTION_CODE_DBID	Promotion Codes
233	PROMOTION_OFFER_DBID	Promotion Orders
234	PROMOTION_VALIDATION_DBID	Promotion Validations
235	WEB_TEMPLATE_DBID	Web Templates
236	PAYMENT_PLAN_DBID	Payment Plans
237	ITEM_CONNECTION_DBID	
238	PAYMENT_PLAN_CONFIG_DBID	Payment Plan Configuration
239	PAYMENT_CONTRACT_STATUS_DBID	Payment Contract Status
240	PLAN_FOP_DBID	Payment Plan Form of Payment
241	RECURRENCE_PATTERN_DBID	
242	ASSOCIATED_TICKET_RATIO_DBID	Associated Ticket Ratio
243	FOP_SET_DBID	Form of Payment Sets
244	FOP_SET_DETAIL_DBID	Form of Payment Set Details
245	STATE_REGION_DBID	
246	WS_HTML_CONNECTION_DBID	
247	COUNTRY_DBID	Countries
248	WS_LOCALIZATION_DBID	Web Store Localizations
249	WS_LOCALIZATION_GROUP_DBID	Web Store Localization Group
250	FOP_DETAIL_DBID	Form of Payment Details
251	DBMS_TAX_DBID	
252	SP_PRICE_SCHEDULE_DBID	Sales Program Price Schedule
253	CATEGORY_PRICE_DATA_DBID	
254	DELIVERY_METHOD_GROUP_DETAIL_DBID	Delivery Method Group Details
255	GX_ITEM_GROUP_DETAIL_DBID	Item Group Details
256	SALES_CHANNEL_NODE_DBID	Sales Channel Node
257	SALES_CHANNEL_PUBLISH_LOG_DBID	Sales Channel Publish Log
258	SP_PRICE_CALENDAR_DBID	Sales Program Price Calendars
259	RESELLER_SALES_CHANNEL_DETAIL_DBID	Reseller Sales Channel Details
260	NUMERIC_RANGE_DBID	
261	DISBURSE_HDR_DBID	Disbursement Headers
262	PROFILE_PRIVS_DBID	User Profile Privileges
263	EGALAXY_TEMPLATE_DBID	eGalaxy Templates
264	CODE_TABLE_DBID	Code Tables
265	CODE_TABLE_VALUE_DBID	Code Table Values
266	CONFIGURATION_OPTIONS_DBID	Central Configuration Options
267	PROFILE_CONTROLS_DBID	User Profile Controls

268	WEB_FARES_DBID	
270	WEB_SCHED_DETAIL_DBID	
271	WS_LOCALIZATION_GROUP_MODULE_DBID	Web Store Localization Group
272	REASONS_DBID	Reasons
273	UPSELL_OPTIONS_DBID	Upsell Options
274	RENEWAL_OPTIONS_DBID	Renewal Options
275	POSTAL_CODE_RANGES_DBID	Postal Code Ranges
276	POSTAL_CODE_RANGE_DETAILS_DBID	Postal Code Range Details
277	FIELD_ATTRIBUTES_DBID	Field Attributes
278	FIELD_ATTRIBUTE_GROUPS_DBID	Field Attribute Groups
279	ITEM_VARIATIONS_DBID	Item Variations
280	PAYMENT_PLAN_PRICE_DATA_DBID	Payment Plan Price Data
281	EGALAXY_HUB_LOG_DBID	eGalaxy Hub Log
282	ITEM_IMAGES_DBID	Item Images
283	PKG_INSTANCE_DBID	Package Instances
284	PKG_INSTANCE_DETAIL_DBID	Package Instance Details
285	IPADDRESS_DBID	IP Addresses
286	SALES_CHANNEL_IPADDRESS_DBID	Sales Channel IP Address
287	IPADDRESS_GROUP_DBID	IP Adress Group
288	TICKET_CANCEL_REASONS_DBID	Ticket Cancel Reasons
289	TERMINAL_ID_DBID	Terminal IDs
290	USER_KEYWORD_DBID	User Keywords
291	PROJECTION_DBID	Projections
292	PROMPT_SECTIONS_DBID	Prompt Sections
293	DBMS_GXFILE_DBID	
294	DBMS_GXFILECONNECTIONS_DBID	
295	SIAE_OPTIONS_DBID	
296	GX_MASTER_KEY_DBID;	
297	ACCESS_CODE_OVERRIDE_DBID	
298	SIAE_VENUES_DBID	
299	SIAE_ITEMS_DBID	
300	SIAE_EVENTS_DBID	
301	SIAE_EVENT_ORGANIZERS_DBID	
302	SIAE_REDUCTION_CODES_DBID	
303	LOYALTY_PROGRAMS_DBID	
304	LOYALTY_ACCRUAL_ITEMS_DBID	
305	LOYALTY_REDEMPTION_ITEMS_DBID	
306	LOYALTY_ACCOUNTS_DBID	
307	GX_PRIVATE_KEY_DBID	
308	POS_CREDIT_TRANSACTION_DBID	
309	SCHED1UDF_DBID	
310	SALES_CHANNEL_LOYALTY_POINTS_DBID	
311	TRANSACTIONAL_UPSELL_OPTIONS_DBID	
312	TRANSACTIONAL_UPSELL_OPTION_REQUIREMENTS_DBID	
313	SIAE_VOID_REASONS_DBID	
314	MULTI_SITE_EXPORT_LOG_DBID	
315	SIAE_CALENDAR_HEADERS_DBID	
316	SIAE_CALENDAR_DETAILS_DBID	
317	PAYMENT_PLAN_RECURRENCE_PATTERN_DBID	
318	ACCESS_CODE_PRINT_DETAIL_DBID	
319	CURRENCIES_DBID	
320	EXPRESSFEERATES_DBID	
321	EXPRESSFEERATEDETAILS_DBID	
322	DESTINATIONTAXES_DBID	
323	EXPRESSCHARGES_DBID	
324	EXPRESSCHARGERULES_DBID	
325	EXPRESSCHARGERULEDETAILS_DBID	
326	EXPRESSCITIES_DBID	
327	EXPRESSSERVICEMODES_DBID	
328	EXPRESSSERVICES_DBID	
329	EXPRESSZONES_DBID	
330	EXPRESSZONEDETAILS_DBID	
331	EXPRESSZONERATES_DBID	
332	EXPRESSZONERATEDETAILS_DBID	
333	EXPRESSDESCRIPTIONS_DBID	
334	EXPRESSSURCHARGES_DBID	
335	EXPRESSSURCHARGEDETAILS_DBID	
336	EXPRESSDEFAULTVALUES_DBID	

337	EXPRESSREQUIREDFIELDS_DBID	
338	RCBT_DBID	
339	MIN_MAX_FARES_DBID	
340	JOINT_MEMBERS_DBID	
341	DISCOUNT_PRIVILEGE_DBID	
342	SALES_CHANNEL_LANGUAGES_DBID	
343	TAX_SETS_DBID	
344	TRANSLATION_FIELDS_DBID	
345	TRANSLATION_LANGUAGES_DBID	
346	TRANSLATION_TABLES_DBID	
347	TRANSLATION_VALUES_DBID	
348	ITEM_TRANSLATIONS_DBID	
349	ATTRACTION_DBID	
350	STOCK_PACKAGES_DBID	
351	MODIFIERS_DBID	
352	MODIFIER_GROUPS_DBID	
353	MODIFIER_GROUPS_DETAIL_DBID	
354	ONLINE_EXCHANGE_DBID	
355	ITEM_MODIFIER_GROUP_DETAILS_DBID	
356	PRICE_PROGRAM_GROUPS_DBID	
357	PASS_PKG_INSTANCE_DBID	
358	COMMUNICATION_METHODS_DBID	
359	GENERIC_CALENDARS_DBID	
360	ATTRIBUTE_VALUE_DBID	
361	ATTRIBUTE_DEFINITION_DBID	
362	GENERIC_CALENDAR_DETAILS_DBID	
363	PLUGIN_ATTRIBUTE_DBID	
364	SALES_CHANNEL_PASSKINDS_DBID	
365	DBMS_PASS_DBID	
366	DBMS_PASS_KIND_GROUP_DETAIL_DBID	
367	ACS_RESERVATION_RULE_SETS_DBID	
368	ACS_RESERVATION_RULE_SETS_ADMIN_HEADERS_DBID	
369	ACS_RESERVATION_RULE_SETS_NOTIFICATIONS_DBID	
370	ACS_RESERVATION_RULE_SETS_CONSEQUENCES_DBID	
371	ACS_RESERVATION_RULE_SETS_STATUSES_DBID	
372	ACS_RESERVATION_RULE_SETS_USER_PROFILES_DBID	
373	ORCA_CARD_PUBLIC_KEYS_DBID	
374	ORCA_BLOCKED_CARDS_DBID	
375	ORCA_RESULT_TYPE_DBID	
376	FARE_PAYMENT_CLOSE_CACHE_DBID	
377	FOP_MENU_HEADERS_DBID	
378	FOP_MENU_DETAILS_DBID	
379	USER_PROFILE_PASS_KIND_GROUPS_DBID	UserProfilePassKindGroups table

## 3.2 Activities

This table stores both Activity Templates and Activities. An Activity is something that is required to happen as a result of a transaction and can be automatically system-generated or manually user-generated. Activity templates are configured for system-generated Activities.

### Columns

Column	Type	Allow Nulls	Description
ActivityID	Int	No	Primary key.
IsTemplate	Bit	No	Indicates whether this activity is a template for system-generated Activities; Default = 0 = False
Status	Int	No	High-level status of the Activity; currently only 2 values: 0 = Open, 1 = Closed; Default = 0
ActivityType	Int	No	Link to CodeTableValue from user-configured lookup table
OwnerID	Int	No	ID of the Work Group or GxUser responsible for the Activity
OwnerType	Int	No	Specifies whether the OwnerID is a 0 = GxUser, 1 = Work Group
State	Int	No	Link to CodeTableValue comes from user-configured lookup table
Frequency	Int	No	Currently 2 values: 0 = Per Item, 1 = Per Transaction; Default = 0
DateOpened	Datetime	Yes	Datetimestamp for when the Activity is created
DateClosed	Datetime	Yes	Datetimestamp for when the Status is changed to 1 = Closed
DaysToComplete	Int	Yes	Optional number of days to be used along with the DateOpened to calculate a CompletionDate
CompletionDate	Datetime	Yes	Optional date for the required completion of this activity - either user selected or calculated based on DaysToComplete
ContactID	Int	Yes	Value not required for an Activity Template, but either ContactID or CustomerID is required for an Activity; Foreign Key to CustContacts table
OrderID	Int	Yes	Foreign key to Orders table
OrderLineID	Int	Yes	Foreign key to OrderLines table
PLU	Char(20)	Yes	Link to PLU in Items table
TranNo	Int	Yes	Link to JnlHeaders.TranNo
NodeNo	Int	Yes	Link to JnlHeaders.NodeNo
PassID	Int	Yes	Link to Passes.PassNo, represents the pass for which this activity was generated
CustomerID	Int	Yes	Value not required for an Activity Template, but either ContactID or CustomerID is required for an Activity; Foreign Key to Customers table

### Indexes

Name	Kind	Columns	Purpose
PKActivitiesActivityID	P	ActivityID	Primary Key.
IXActivitiesOwnerTypeOwnerID	IX	OwnerType, OwnerID	For selection by OwnerType and OwnerID
IXActivitiesContactID	IX	ContactID	For selection by Customer Contact
IXActivitiesPLU	IX	PLU	For selection of Activities by PLU
IXActivitiesStatusAndOwner	IX	Status, IsTemplate, OwnerType, OwnerID	Used when querying for user Activity statistics for display on main menu.
IXActivitiesOrderID	IX	OrderID	For selection by Order ID
IXActivitiesTranNoNodeNo	IX	TranNo, NodeNo	For selection by transaction number, node number

### 3.3 ActivityNotes

This table stores notes related to an Activity.

#### Columns

Column	Type	Allow Nulls	Description
ActivityNoteID	Int	No	Primary Key.
ActivityID	Int	No	Foreign key to Activities table.
CreateDate	Datetime	No	Datetime for when the note was created
UserID	Int	No	Link to GxUsers.UserID; the ID of the last user to modify the note
Note	Text	Yes	The content of the note.

#### Indexes

Name	Kind	Columns	Purpose
PKActivityNoteID	P	ActivityNoteID	Primary Key.

### 3.4 Agencies

This table stores Galaxy's agencies information.

#### Columns

Column	Type	Allow Nulls	Description
AgencyUniqueID	Int	N	Primary key, always unique. System generated.
AgencyNo	Int	N	User definable agency number. This is the value used to identify an agency throughout the system.
AgencyName	VarChar(100)	Y	Name to identify the AgencyNo for display purposes
Files	Char(20)	Y	
Directory	Char(64)	Y	
AgencyDescription	VarChar(100)	Y	
DefaultMenuID	Char(16)	Y	
DefaultResourceID	Int	Y	A default Resource ID to use when an Event ticket was selected at POS for this Agency
AgencyType	Int	Y	Indicates the type of agency. Can be used to determine where in the system the agency can be selected. <sup>1</sup>
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>2</sup>
AgencyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKAgencyUniqueID	P	AgencyUniqueID	Primary key.

#### <sup>1</sup> AgencyType Values

Gateway Constant Name	Value	Description
AGENCY_TYPE_DEFAULT	0	Identifies a Galaxy Agency
AGENCY_TYPE_RESELLER	1	Identifies a Reseller Agency

#### <sup>2</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore Agency records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store Agencies.

### 3.5 AgencyControls

This table stores references that associate agencies to nodes, profiles and agencies to item groups. It defines what agencies any given profile can logon to, what item groups can be sold at a given agency and the nodes that are in an agency.

#### Columns

Column	Type	Allow Nulls	Description
AgencyControlID	Int	N	Primary key, always unique. System generated.
AgencyNo	Int	N	Agency number
AuxID	Char(20)	N	Item Group or profile ID number
Kind	Int	N	Specifies what table the AuxID column references <sup>1</sup>
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>2</sup>
AgencyControlGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKAgencyControlAgencyControlID	P	AgencyControlID	Primary key.

#### <sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	AGENCY_CONTROL_KIND_USER_PROFILE	AuxID references a user profile
2	AGENCY_CONTROL_KIND_ITEM_GROUP	AuxID references an item group
3	AGENCY_CONTROL_KIND_NODE	AuxID references a node number
4	AGENCY_CONTROL_KIND_USER	AuxID references a user

#### <sup>2</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore Agency Control records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store Agency Controls.

### 3.6 AgencyNodeRanges

This table Stores the available node ranges allowed for a given reseller agency.

#### Columns

Column	Type	Allow Nulls	Description
AgencyNodeRangeID	Int	N	Primary key, always unique. System generated.
AgencyID	Int	N	Agency number
FromNodeNo	Int	N	Start value of the node range
ThruNodeNo	Int	N	End value of the node range

#### Indexes

Name	Kind	Columns	Purpose
PKAgencyNodeRangeID	P	AgencyNodeRangeID	Primary key.
IXAgencyNodeRangeAgencyID		AgencyID	Used by query to retrieve all node ranges for a given AgencyID

### 3.7 AssociatedTicketRatios

Stores the Associated Ticket PLU and associated Item Group or PLU information assigned to customer categories.

#### Columns

Column	Type	Allow Nulls	Description
AssociatedTicketRatioID	Int	N	Primary key, always unique
CustCategoryID	Int	N	Customer category ID to which the Item Group AssociatedTicket set is associated with.
ItemGroupID	Int	N	Points to the associated Item Group associated with the AssociatedTicketPLU
Ratio	Int	N	Stores the ration of associated tickets needed per guest count.
AssociatedTicketPLU	Varchar(40)	N	PLU of the ticket we will associate with an item group or PLU
GuestPLU	Varchar(40)	N	PLU associated with AssociatedTicketPLU
ApplyMethod	Int	Y	Indicates which method to use when calculating ratios
AssociatedTicketRatioGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKAssocTktRtiosAssocTktRti0D	P	AssociatedTicketRatioID	Primary Key.

### 3.8 BadChecks

The purpose of the table is to store all Blacklisted checking accounts. Any checking account in this data will be rejected as a valid form of payment.

#### Columns

Column	Type	Allow Nulls	Description
BadCheckID	Int	N	Primary key, always unique
RoutingID	Char(9)	N	Bank routing number
AccountID	Char(18)	N	Customer Bank account number
SequenceNumber	Int	N	Sequence number of the import file.

#### Indexes

Name	Kind	Columns	Purpose
PKBadCheckID	P	BadCheckID	Primary Key.
IXBadChecksAccountIDRoutingID		RoutingID, AccountID	To allow queries by AccountID and RoutingID

### 3.9 BinDetails

Stores demographic details about BINs for card numbers that were used in Galaxy transactions. This information is populated by the BIN Lookup Service, which looks up this information in the third-party BINDB API.

#### Columns

Column	Type	Allow Nulls	Description
BinDetailID	Int	N	Primary key, always unique
Bin	Char(6)	N	The BIN for this record
Brand	NVarChar(128)	Y	The card company/vendor which issued this card, eg., Visa/MasterCard/etc.
BankName	NVarChar(128)	Y	Name of bank of financial institution which issued the card, eg., Citibank
CardType	NVarChar(32)	Y	Type of card, eg., Credit/Debit
Level	NVarChar(32)	Y	"Level" of card, eg., Platinum/Gold/Business/etc.
IsoCountry	VarChar(64)	Y	Full name of country of origin for card, eg., "United States"
Info	NVarChar(128)	Y	Additional information about card from BINDB
CountryIso	Char(2)	Y	Two letter ISO country code, eg., "US"
Country2Iso	Char(3)	Y	Three letter ISO country code, eg., "USA"
Country3Iso	Char(3)	Y	Numeric ISO country code, eg., '840'
WebAddress	VarChar(128)	Y	URL for issuing bank's main website
Phone	VarChar(128)	Y	Contact phone for issuing bank
LastRetrieved	DateTime	Y	Date/time this record was last retrieved from BINDB
BinDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKBinDetailID	P	BinDetailID	Primary key

### 3.10 BINs

For the system to automatically validate the validity of the type of bank card being used, it is necessary to define what numbers are used by each card type. The **Bins** table contains the these bank card formats.

#### Columns

Column	Type	Allow Nulls	Description
BinID	Int	N	Primary key, always unique
Name	Char(16)	Y	The name of the credit card being defined, i.e. VISA.
LoPrefix	Char(12)	Y	The lowest number of the credit card's prefix. For example, if the first two digits of the credit card must be from 51 to 55, enter 51 in the Lo Prefix field.
HiPrefix	Char(12)	Y	The highest number of the credit card's prefix. For example, if the first two digits of the credit card must be from 51 to 55, enter 55 in the Hi Prefix field.
BinLength	Int	Y	The total length in digits of the credit card number.
FOP	Int	Y	This number links the credit card number with its Protocol and Terminal ID, through the Form of Payment definition.
Mod10Check	Bit	N	If TRUE, the Mod 10 method of card number validation is used.
DateCheck	Bit	N	If TRUE, the card's expiration date will be validated.
User1	Char(16)	Y	The fields User1 through User4 are used by the Gateway's BCAM2 system. For local credit authorization, these fields are blank.
User2	Char(16)	Y	As noted above
User3	Char(16)	Y	As noted above
User4	Char(16)	Y	As noted above
Abbr	Char(8)	Y	An abbreviation of the credit card name, such as its corresponding Form of Payment name.
IatanCard	Bit	N	This field only applies to the IATA Network travel agent card. For most credit card processors, this field is left blank.
ServiceCode	Char(4)	Y	The Service Code field is used when it is necessary to attach a specific service code to a credit card type. For example, debit cards are usually rejected if they do not include a service code of 120. For a debit card type, the service code field would be set to 120.
IssuingCompanyCode	Char(12)	Y	Alphanumeric to identify the credit card issuing company.
CVNBitmapFileName	VarChar(40)	Y	Indicates the filename of an image to display when the system prompts for entry of a card verification number (for a manually entered credit card corresponding to the record's card range).
SVTypeID	Int	Y	FK reference to DebitTypes.DebitTypeID. Indicates the stored value type record that is linked to this BIN.
ConstructCardNo	Bit	Y	Defines whether there are related entries in the MediaIDDetails table defining how to construct the Card No.
MediaType <sup>1</sup>	Int	Y	Defines which track to use when constructing the Card No.
BinGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKBinBinID	P	BinID	Primary key.

#### <sup>1</sup> MediaType Values

Value	Gateway Constant Name	Description
0	ANY_MEDIA_TYPE	Use any media to construct Card No
1	TRACK1_MEDIA_TYPE	Use track 1 to construct Card No
2	TRACK2_MEDIA_TYPE	Use track 2 to construct Card No
3	SCAN_MEDIA_TYPE	Use scan to construct Card No

### 3.11 BlockoutDates

BlockoutDates table contains the block out dates that can be applied to dated ticket types. A block out date can be a simple date range or it can be relative to the sale date.

#### Columns

Column	Type	Allow Nulls	Description
BlockoutDateID	Int	N	Primary key, always unique
BlockoutID	Int	N	ID of the Blockout. Assigned by Galaxy when the Blockout is created locally
Description	Varchar(30)	Y	Description explaining the block out
Basis	Int	N	Basis for the blockout See Basis values below <sup>1</sup>
StartDateTime	Datetime	Y	Blockout Start date time. Only applies for basis of bbAbsolute.
EndDateTime	Datetime	Y	Blockout End date time. Only applies for basis of bbAbsolute.
StartYears	Int	Y	Blockout start year. Applies for basis of bbRelative and bbMonths.
StartMonths	Int	Y	Blockout start months. Applies for basis of bbRelative and bbMonths.
StartDays	Int	Y	Blockout start days. Applies for basis of bbRelative and bbMonths.
EndYears	Int	Y	Blockout end year. Applies for basis of bbRelative and bbMonths.
EndMonths	Int	Y	Blockout end months. Applies for basis of bbRelative and bbMonths.
EndDays	Int	Y	Blockout end days. Applies for basis of bbRelative and bbMonths.
BlockoutDateGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKBlockoutDateID	P	BlockoutDateID	Primary Key.
IXBlockoutDatesBlockoutID		BlockoutID	Used by query to retrieve the blockout definition for a given BlockoutID

#### <sup>1</sup> Basis Values

Value	Gateway Constant Name	Description
0	bbAbsoluteDate	Blockout dates are defined as from and thru range
1	bbRelativeToDate	Blockout dates are relative to the sale date
2	bbRelativeMaintainDayOfMonth	Blockout dates are relative to the sale date with day in the month maintained

### 3.12 BlockoutGroupDetails

BlockoutGroupDetails table contains the block out dates associated with a given blockout group. In Galaxy it is possible to group the blockout dates (possibly applicable to same ticket type) into one group and then the entire blockout group is assigned to a ticket type.

#### Columns

Column	Type	Allow Nulls	Description
BlockoutGroupDetailID	Int	N	Primary key, always unique
GroupID	Int	N	Id of the Blockout Group. Assigned by Galaxy when the Blockout Group is created locally. FK reference to BlockoutGroups.ID column.
BlockoutID	Int	N	ID of the Blockout. Assigned by Galaxy when the Blockout is created locally. FK reference to BlockoutDates.BlockoutID column.
DetailID	Int	Yes	This is the unique ID in the local data - used by the central database download process to local the local record
BlockoutGroupDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKBlockoutGroupDetailID	P	<i>BlockoutGroupDetailID</i>	Primary Key.
IXBlockoutGroupDetailGroupID		<i>GroupID</i>	Used by query to retrieve the blockout definition for a given GroupID

### 3.13 BlockoutGroups

The BlockoutGroups table contains user-defined groups that are associated with specific PLUs, to be used for date-specific tickets.

#### Columns

Column	Type	Allow Nulls	Description
BlockoutGroupID	Int	N	Primary key, always unique
ID	Int	Y	ID matching with the equivalent B-Tree records
Description	Char(30)	Y	The description used for this blockoutgroup
BlockoutGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKBlockoutGroupID	P	BlockoutGroupID	Primary Key.

### 3.14 BonusPromotions

The BonusPromotions table contains promotions that are configured to award bonus points or discounts when selected in a loyalty card transaction.

#### Columns

Column	Type	Allow Nulls	Description
BonusPromotionID	Int	N	Primary key, always unique.
Description	VarChar(60)	Y	Describes the bonus promotion
PromotionType	Int	N	Specifies the type of promotion benefit <sup>1</sup> .
Value	Float	N	Meaning depends upon PromotionType field <sup>1</sup> .
ApplyMethod	Int	Y	Describes how the system will apply this promotion if eligible <sup>2</sup> .
MaxBonusAmount	Float	Y	The maximum number of bonus points that may be awarded.
RestrictItemGroup	Int	Y	Foreign key, reference to GxItemGroups.ItemGroupID. If non-zero, specifies the group of items this promotion may apply to.
RequireItemGroup	Int	Y	Foreign key, reference to GxItemGroups.ItemGroupID. If non-zero, specifies the items that can be used to enable this promotion.
ItemPLU	VarChar(20)	Y	Foreign key, reference to Items.PLU. If specified, indicates the item to select upon application of this promotion.

#### Indexes

Name	Kind	Columns	Purpose
PKBonusPromotionID	P	PromotionID	Primary Key.

#### <sup>1</sup> PromotionType Values

Value	Gateway Constant Name	Description
1	PROMOTION_TYPE_AMOUNT	Value specifies an amount of bonus points to award.
2	PROMOTION_TYPE_PERCENT	Value specifies a percentage of bonus points to award, relative to the number of points that would have been issued anyway (for example, 200.0 would award a total of triple points).
3	PROMOTION_TYPE_DISCOUNT	Value specifies a discount to apply (instead of awarding bonus points).

#### <sup>2</sup> ApplyMethod Values

Value	Gateway Constant Name	Description
0	APPLY_METHOD_AUTOMATIC	The system will automatically select the promotion. It can be deselected, if desired, by using the "Bonus Promotions Summary" function.
1	APPLY_METHOD_CONFIRM_NO	The system will prompt for whether or not the promotion should be selected. By default, it will not be selected.
2	APPLY_METHOD_CONFIRM_YES	The system will prompt for whether or not the promotion should be selected. By default, it will be selected.
3	APPLY_METHOD_MANUAL	By default, the system will not select this promotion, nor will it prompt for it (unless it needs to prompt for another promotion anyway). The "Bonus Promotions Summary" function can be used to manually select this promotion.
4	APPLY_METHOD_INACTIVE	The system will not allow selection of this promotion, similarly as if the promotion was not in the database at all.

### 3.15 CancelReasons

The CancelReasons table contain reasons that can be used for the Galaxy ticket cancel function.

#### Columns

Column	Type	Allow Nulls	Description
CancelReasonID	Int	N	Primary key, always unique.
ShortDescription	Char(16)	N	Short description to identify the cancel reason.
Reason	Char(80)	N	The reason for the ticket cancellation.
ReasonActive	Bit	N	

#### Indexes

Name	Kind	Columns	Purpose
PKCancelReasonID	P	CancelReasonID	Primary key.

### 3.16 Carriers

The Carriers table contains records of Companies providing intercity or other scheduled service suitable for using the Quasar routing and scheduling engine.

#### Columns

Column	Type	Allow Nulls	Description
CarrierID	Integer	N	Primary Key
Carrier	Char(4)	N	Abbreviation - generally 3 chars - used as the primary identifier for a carrier, for example CARRIER
CompanyNumber	Smallint	Y	Galaxy company number
Name	VarChar(60)	Y	Full company name
Street1	Char(30)	Y	
Street2	Char(30)	Y	
City	Char(30)	Y	
State	Char(40)	Y	
Zip	Char(16)	Y	
Country	Char(20)	Y	
Contact1	Char(30)	Y	
Phone1	Char(30)	Y	
Contact2	Char(30)	Y	
Phone2	Char(30)	Y	
URL	VarChar(80)	Y	
Fax	Char(30)	Y	
Tax1_ID	Char(16)	Y	
Tax2_ID	Char(16)	Y	
Imported	Bit	Y	Used when importing TX data to delete missing records

#### Indexes

Name	Kind	Column	Purpose
PKCarriersCarrierID	P	CarrierID	Primary Key
IXCarriersCarrier	I	Carrier	Index on the Carrier column

### 3.17 CCFTables

List of all the tables that have been included for a given CCF.

#### Columns

Column	Type	Allow Nulls	Description
CCFTableID	Int	N	Primary key, always unique
CCFStatusID	Int	N	Link to CCFStatuses Table
TableID <sup>1</sup>	Int	N	Numeric ID of the table selected
TableAction <sup>2</sup>	Int	N	Add, Delete, Revise, or Overwrite
FileFormat <sup>3</sup>	Int	N	CCFLegacy, CSV. Format the exported data is in.

#### Indexes

Name	Kind	Columns	Purpose
PKCCFTablesCCFTableID	P	CCFDBid	Primary Key.

#### <sup>1</sup> Table ID Values

See Table ID information as Indexed at the beginning of this document.

#### <sup>2</sup> Table Action Values

Value	Gateway Constant Name	Description
0	NONE	None
1	ACTION_ADD	Add
2	ACTION_REVISE	Revise
3	ACTION_DELETE	Delete
4	ACTION_OVERWRITE	Overwrite

#### <sup>3</sup> File Format Values

Value	Gateway Constant Name	Description
1	LEGACYCCF	Original CCF format
2	CSV	CSV
3	XML	XML
4	SQL_BULK	SQL Bulk Insert

### 3.18 CCFNodes

Records and tracks the status of each node that is to receive the CCF.

#### Columns

Column	Type	Allow Nulls	Description
CCFNodeID	Int	N	Primary key, always unique
CCFStatusID	Int	N	Link to CCFStatuses table
Node	Int	N	Node number
Status <sup>1</sup>	Int	N	Current status - Waiting, Sent, Received, Processed, Error
StatusMessage	Char(30)	Y	System Error Message
FileName	Char(30)	N	Filename of the CCF

#### Indexes

Name	Kind	Columns	Purpose
PKCCFNodesCCFNodeID	P	CCFNodeID	Primary Key.

#### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	NODE_WAITING	Waiting to receive CCF
2	NODE_SENT	CCF has been sent to the node
3	NODE_RECEIVED	Received CCF
4	NODE_PROCESSED	CCF has been processed
5	NODE_ERROR	An error has occurred. See Status Message for details.

### 3.19 CCFStatuses

This table is used to track the current status of a given CCF. The data provided here can be viewed via a CCF maintenance facility.

#### Columns

Column	Type	Allow Nulls	Description
CCFStatusID	Int	N	Primary key, always unique
FileName	Char(30)	N	CCF Filename
CreateDate	DateTime	N	Date/Time CCF was created
EffectiveDate	DateTime	N	Date/Time CCF is to be applied
SentDate	DateTime	Y	Date/Time CCF was sent to all the nodes
Status <sup>1</sup>	Int	N	Status code of the CCF

#### Indexes

Name	Kind	Columns	Purpose
PKCCFStatusesCCFStatusID	P	CCFStatusID	Primary Key.

#### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	STATUS_SEND	CCF has been created but not sent out.
2	STATUS_SENT	CCF has been sent to all the nodes
3	STATUS_COMPLETE	CCF has been created, sent, received, and processed by all nodes.

### 3.20 ChangeFundOverride

This table provides change fund values for specific User ID's that override those defined for each POS node.

#### Columns

Column	Type	Allow Nulls	Description
ChangeFundOverrideID	Int	N	Primary key, always unique. This is the ID of a particular user (or "agent")
Amount	Money	N	Currency amount of the override
Used	Bit	N	Flag to indicate whether this override is being used

#### Indexes

Name	Kind	Columns	Purpose
PKChangeFundOverrideChgFndOvID	P	ChangeFundOverrideID	Primary key.

### 3.21 ChangeFund

This table provides current and default change fund values by shift for individual POS nodes.

#### Indexes and Constraints

Primary Key: PKChangeFundID

Indexes:

(None)

Column	Type	Allow Nulls	Description
ChangeFundID	Int	N	Primary key, always unique. This is the number of a particular node
Shift1Default	Money	N	Default change fund amount for shift number 1
Shift2Default	Money	N	Default change fund amount for shift number 2
Shift3Default	Money	N	Default change fund amount for shift number 3
Shift4Default	Money	N	Default change fund amount for shift number 4
Shift5Default	Money	N	Default change fund amount for shift number 5
Shift1Current	Money	N	The current change fund amount for shift number 1
Shift2Current	Money	N	The current change fund amount for shift number 2
Shift3Current	Money	N	The current change fund amount for shift number 3
Shift4Current	Money	N	The current change fund amount for shift number 4
Shift5Current	Money	N	The current change fund amount for shift number 5

## 3.22 CITIES

The Cities table is used by the Transportation module and Web Store for Transportation.

### Columns

Column	Type	Allow Nulls	Description
CityID	Integer	N	Primary Key
Code	Integer	Y	Industry standard ID number
Owner	Char(4)	Y	Refers to Carriers.Carrier
Name	Char(24)	Y	City name - the number of characters is limited by a) real estate on ticket stock and b) reports and display space in TX which used 14 characters for the city name plus 2 for state as in READING/PA (16 chars total)
LongName	VarChar(40)	Y	Full city name
State	Char(2)	Y	State/Provence
Abbr	Char(4)	Y	4 char abbreviation (limited to 3 to match Greyhound)
ZipCode	Char(16)	Y	Also known as postal code in Canada
TimeZone	Char(1)	Y	NAECMPK
DST	Bit	Y	Is Daylight Saving Time used at this location Y/N
PTP	Smallint	Y	Preferred Transferr Point rating from 0 to 10
MinCT	Smallint	Y	Default Minimum Connect Time
MaxCT	Smallint	Y	Defaut Maximum Connect Time (not currently honored)
FID	Char(12)	Y	Feature ID in the USGS database
Latitude	Float	Y	In decimals
Longitude	Float	Y	In decimals
Imported	Bit	Y	Used when importing TX data to delete missing records

### Indexes:

Name	Kind	Column	Purpose
PKCitiesCityID	P	CityID	Primary Key
IXCitiesAbbr	I	Abbr	Index on Abbr column
IXCitiesCode	I	Code	Index on Code column
IXCitiesLatitude	I	Latitude	Index on Latitude column
IXCitiesLongitude	I	Longitude	Index on Longitude column
IXCitiesStateName	I	State, Name	Index on State and Name columns

### 3.23 COA

The Chart of Accounts (COA) is the central configuration in the Galaxy system. Every company has its own Chart of Accounts, and every ticket, form of payment, and tax amount is recorded by its account number in the journal when a transaction is completed. The COA is the account into which the money goes, and is similar to a general ledger in accounting but the COA does not track expenses. Set up your COA in a way that agrees with your accounting practices.

#### Columns

Column	Type	Allow Nulls	Description
COAID	Int	N	Primary key, always unique. System generated.
AccountID	Varchar(12)	N	A string representation of the entire COA record, which includes CompanyID (4 digits), GLCode (3 digits), Category (3 digit) and SubCat (2 digits). Ex '001053200302'
CompanyID	Int	Y	The company this COA entry is for, FK to Companies.CompanyID
GLCode	Int	N	General Ledger code <sup>1</sup>
Category	Int	N	<p>These two fields are subject to user input and can consist of any number sequence. Rules exist for certain account numbers.</p> <p>The Chart of Accounts is always organized in numeric order. This means that account numbers entered may be organized numerically but this may not be the intended organization.</p> <p>Taxes, Forms of Payment, and Deposits have only certain possible values for their Category and Sub-category. For taxes, the category is the only valid field and is numbered according to the sales tax number being defined. For example, if the account number is being defined for tax number 1, the account number is 120-001-00. For Forms of Payment, the category and sub-category correspond to the two digit form of payment code (see Edit Forms of Payment for more information). For example, if the payment code is 23, the account number is 532-002-03.</p>
SubCat	Int	N	
Name	Varchar(35)	N	The label for the account number to be displayed when reports are generated. Depending on the Report Action, when this label is displayed in the list of account numbers within the system, it is automatically indented.
AccountKind	Char(1)	N	<p>The current account number, one of the following options:</p> <ul style="list-style-type: none"> <li>• <i>Summary</i> - Indicates that the account number summarizes other account numbers.</li> <li>• <i>Detail</i> - Indicates that the account number that posts actual values.</li> </ul>
RptAction	Int	N	<p>The level of detail you wish to display on a report.</p> <ul style="list-style-type: none"> <li>• <i>Levels 1 through 4</i> - control sub-totalling on the Income Statement. Level 1 is considered to be the most amount of detail on the report. Level 4 is the least amount of detail. As a general rule, Level 1 account numbers are almost always detail account kind and Level 4 account numbers are almost always summary account kind.</li> <li>• <i>Disappear</i> - suppresses printing on the Income Statement.</li> </ul>
CommissionRate	Float	N	The commission rate to be applied for this account. This amount is recorded in each ticket record and listed on the Commission Report. Commission amounts are only calculated for account numbers with a code of 101.
UserCode1	Varchar(10)	N	Used to match general ledger account numbers to the system's COA. Use the Extended Code Journal Export function to create sales reports based on general ledger numbers.
UserCode2	Varchar(10)	N	
UserCode3	Varchar(10)	N	
CodeDescr	Varchar(8)	N	The label for the account number to be displayed when viewing the sales journal. Because of space restraints in this view, this field is limited to eight characters.
AmtAction	Char(1)	N	How the amount field for this account is summarized on the income statement in the summary accounts above it. The available options are <b>Add</b> , <b>Subtract</b> and <b>Ignore</b> .
QtyAction	Char(1)	N	How the quantity field for this account is summarized on the income statement in the summary accounts above it. The available options are <b>Add</b> , <b>Subtract</b> and <b>Ignore</b> .
CommissionKind	Int	N	<p>The commission on a ticket can be calculated based on the following values for a ticket:</p> <ul style="list-style-type: none"> <li>• <i>Net Price</i> - The value that is actually posted to the account number. It does not include any tax values and is after any discounts have been deducted.</li> <li>• <i>Discounted Price</i> - This is the value of the ticket including tax after discounts have been deducted.</li> <li>• <i>Gross Price</i> - This is the value of the ticket including tax and before discounts have been deducted.</li> </ul>
ActiveInd	Char(1)	N	Always 'Y'
Inactive	Bit	N	True if COA is Inactive, and not visible in most picklists.

#### <sup>1</sup> GLCode Values

Value	Gateway Constant Name	Description
101	TKT_REC	Ticket record
102	ITEM_REC	Item record
103	FEE_REC	Fee record
104	DONATION_REC	Donation record
120	TAX_REC	Tax record
210	PAIDIN_REC	Paid-in record
310	PAIDOUT_REC	Paid-out record
401	TKT_REFUND_REC	Ticket refund record
420	ST_TAX_REFUND_REC	Tax refund record
532	PAYMENT_REC	Payment record
533	REISSUE_REC	Reissuance information for a transportation ticket.
610	DEPOSIT_REC	Deposit record
620	COA_CHANGE_REC	Starting change fund for the shift with COA info

#### Indexes

Name	Kind	Columns	Purpose
PKCOACOAI	P	COAID	Primary key

### 3.24 CodeTables

This table holds all the information pertaining to a Code Table.

#### Columns

Column	Type	Allow Nulls	Description
CodeTableID	Int	N	Primary key, always unique
Name	VarChar(30)	N	Name of Code Table
Description	VarChar(256)	N	Description of the Code Table
Active	Boolean - Bit	N	Is the Attribute Active. 0 = Active, 1 = Inactive
CodeTableGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKCodeTablesCodeTableID	P	CodeTableID	Primary Key.
IXCodeTablesCTblIDName	IX	CodeTableID, Name	Unique Index

### 3.25 CodeTableValues

This table holds all the information pertaining to a CodeTable Value.

#### Columns

Column	Type	Allow Nulls	Description
CodeTableValueID	Int	N	Primary key, always unique
CodeTableID	Int	N	Links Detail to Code Table
Value	VarChar(256)	N	The value for this entry
Description	VarChar(256)	N	Description of the Code Table Value
Sequence	Int	N	The sequence number of the entry used for ordering values in a list.
Mask	VarChar(256)	Y	Formatting for the Value
Active	Boolean - Bit	N	Is the Attribute Active. 0 = Active, 1 = Inactive
CodeTableValueGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKCTVCodeTableValueID	P	UserSelectionDetailID	Primary Key.
IXCodeTableValuesCTblIDValue		CodeTableID, Value	Unique index to prevent users from defining the same value for a given table.

#### <sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
0	usdtString	TuserSelectionDetailType = (usdtString, usdtInteger);
1	usdtInteger	TuserSelectionDetailType = (usdtString, usdtInteger);

### 3.26 CodeValues

#### Columns

Column	Type	Allow Nulls	Description
CodeValueID	Int	N	Primary key, always unique
CodeName	Varchar(50)	N	
CodeValue	Varchar(10)	N	
TypeCode	Varchar(10)	N	
EntryType	Char(1)	N	
ActiveInd	Char(1)	N	

#### Indexes

Name	Kind	Columns	Purpose
(none)			

#### <sup>1</sup> CodeValueID Values

Values	Const Name	Journal code value for
1	HDR_REC	Header record
4	VOID_REC	Void record
5	REFUND_REC	Refund record
7	ID_MEMO_REC	ID Memo record
8	POLL_REC	Poll record
9	MEMO_REC	Memo record
10	LOCK	Lock record
11	UNLOCK	Unlock record
12	LOGON	Logon record
13	LOGOFF	Logoff record
14	CASHOUT	Cashout record
15	MAINT	User selected maintenance mode
16	ONLINE	User selected ticketing mode
17	RESTART	System restarted after a power-failure or crash
18	STARTUP	System Startup
21	DOS_MODE	Exited to DOS
22	ACCOUNT_MODE	User selected accounting mode
23	NETWORK_MODE	User selected network config/operation
24	CUST_ID_REC	Customer ID record
27	OVER_SHORT	Over/Short amount for the shift
28	STOCK_ISSUE	Ticket stock issued to booth
29	STOCK_USAGE	Ticket stock usage, registered at cashout
30	STOCK_RETURN	Ticket stock returned from booth
31	STOCK_VOID	Ticket stock void, registered at cashout
32	STOCK_V_RET	Ticket stock returned voided at admission
33	CHANGE_REC	Starting change fund for the shift.
34	DISNEY_REC	Disney dollars-Entered at end of shift
35	DRAFT_REC	Credit authorization information
37	ORDER_REC	Order transaction record
38	EVENT_REC	Event transaction record
39	OLD_USAGE_REC	Usage record
40	INVOICE_REC	Invoice record
41	SETTLE_REC	Settlement record
42	CURRENCY_REC	Foreign currency record
43	CAPACITY_REC	Event capacity record
44	RES_REAT_REC	Reserved seat record
45	OLV_DRAFT_REC	Additional draft data
46	CCF_MEMO_REC	CCF memo data
47	VISUAL_ID_REC	Visual ID record
48	RESERVATION_REC	Resource reservation
49	RENTAL_REC	Rental ticket
50	NEW_USAGE_REC	New usage record
51	VOUCHER_REC	Voucher record
52	RECEIPT_REC	Receipt printed on demand
53	EXTRA_TKT_INFO_REC	Extra ticket information record
54	REASON_REC	Reason record
55	SURVEY_REC	Survey response record
56	DENOMINATION_REC	Denominational deposit record
57	SUSPEND_SHIFT_REC	Current shift has been suspended for remote cashout
58	EXTRA_ITEM_INFO_REC	Transactional tax records for items

59	CASHOUT_SHIFT_REC	
60	NO_SALE_REC	No sale, open cashdrawer transaction
61	DEBIT_REC	Debit sale transaction
62	RECHARGE_REC	Recharge to debit
63	SALES_PROG_REC	Reference to any sales program used
101	TKT_REC	Ticket record
102	ITEM_REC	Item record
120	TAX_REC	Tax record
210	PAIDIN_REC	Paid-in record
310	PAIDOUT_REC	Paid-out record
401	TKT_REFUND_REC	Ticket refund record
420	ST_TAX_REFUND_REC	Tax refund record
532	PAYMENT_REC	Payment record
610	DEPOSIT_REC	Deposit record
620	COA_CHANGE_REC	Starting change fund for shift. This record contains COA info.

### 3.27 Companies

The Galaxy Point of Sale system, although a system for automating the sale of tickets, retail and food, is also, at its core, a package for reporting the revenue from those sales. At its core, the company file is used to build the structure for accounting for those sales. A company is the main operating company, a separate or subsidiary company, or a tenant company where tickets are sold for a commission. Sales reports are printed with all companies consolidated or with a separate report for each individual company.

#### Columns

Column	Type	Allow Nulls	Description
CompanyUniqueID	Int	N	Primary key, always unique. System generated.
CompanyID	Int	N	This field contains a user definable three-digit identifier for the company. The Company ID is used in account numbers throughout Galaxy.
Name	nvarchar(26)	Y	A long name for the company.
Abbr	nchar(4)	Y	A short abbreviation for the company.
FOPMask	nvarchar(8)	Y	The FOP to be used for each company. Not used by the system.
Serial	Int	Y	Current serial number.
LastSerial	Int	Y	The serial number to roll over.
ExternalID	nvarChar(8)	Y	External identifier for a company. Ex: Used by Micros FIAS for sales outlet, which is the equivalent to the Galaxy company.
FiscalID	nvarchar(20)	Y	The fiscal ID for the company. A fiscal ID could be a government issue number like a SIRET# for France.
Street1	nvarchar(255)	Y	Street address, first line.
Street2	nvarchar(255)	Y	Street address, second line.
Street3	nvarchar(255)	Y	Street address, third line.
City	nvarchar(40)	Y	City.
State	nvarchar(20)	Y	State or province.
Postal	nchar(16)	Y	Zip / Postal code.
CountryCode	nchar(2)	Y	Country Code.

#### Indexes

Name	Kind	Columns	Purpose
PKCompanies	P	CompanyUniqueID	Primary key.

### 3.28 ConfigurationOptions

This table holds centralized configuration options.

#### Columns

Column	Type	Allow Nulls	Description
ConfigurationOptionID	Int	N	Unique Identifier
Description	VarChar(100)	N	Name for the configuration option
Value	VarChar(1000)	N	Value for this configuration option, converted (if necessary) and stored as a string
DataType	Int	N	Type of the configuration option <sup>1</sup>
Code	Int	N	Constant associated with a particular central configuration setting <sup>2</sup>
Scope	Int	Y	Specifies the scope in which this configuration option applies to. <sup>3</sup>
OwnerId	Int	N	Specifies the specific owning entity if scope is agency or node. If scope is agency, OwnerId will reference an agency. If scope is node, OwnerId will reference a node.
ConfigurationOptionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKConfigurationOptionID	P	ConfigurationOptionID	
IXConfigurationOptionsCode	IX	Code	Improve query performance

#### <sup>1</sup> DataType Values

Value	Gateway Constant Name	Description
1	INTEGER_DATA_TYPE	Value is an Integer
2	STRING_DATA_TYPE	Value is an String
3	BOOLEAN_DATA_TYPE	Value is a Boolean (Stored as YES/NO)
4	DATE_TIME_DATA_TYPE	Value is a DateTime (Stored as YYYYMMDD HH:MM using military time)
5	DATE_DATA_TYPE	Value is a Date (Stored as YYYYMMDD)
6	TIME_DATA_TYPE	Value is a Time (Stored as HH:MM using military time)
7	CURRENCY_DATA_TYPE	Value is a Currency
8	REAL_DATA_TYPE	Value is a Real/Double

#### <sup>2</sup> Code Values

Value	Gateway Constant Name	Description
0	CC_CODE_INVALID	[ Invalid Code ]
1	NOT USED	NOT USED
2	NOT USED	NOT USED
3	CC_CODE_MIN_PASSWORD_LENGTH	Users - Security - Minimum Password Length
4	NOT USED	NOT USED
5	CC_CODE_NUM_PASSWORDS_TO_REMEMBER	Users - Security - Passwords to Remember
6	CC_CODE_MAX_PASSWORD_AGE	Users - Security - Maximum Password Age in days
7	NOT USED	NOT USED
8	CC_CODE_REQUIRE_UPPER_ALPHA_CHAR	Users - Security - Require uppercase alphabetic characters
9	CC_CODE_REQUIRE_LOWER_ALPHA_CHAR	Users - Security - Require lowercase alphabetic characters
10	CC_CODE_REQUIRE_INSET_NUMERALS	Users - Security - Require numerals in positions other than the end
11	NOT USED	NOT USED
12	CC_CODE_MAX_EXPIRED_LOGONS	Users - Security - Max Expired Logons
13	NOT USED	NOT USED
14	CC_CODE_MAX_FAILED_LOGON_ATTEMPTS	Users - Security - Max failed logon attempts
15	CC_CODE_INACTIVITY_LOCKOUT_DAYS	Users - Security - Inactivity lockout in days
16	NOT USED	[not currently used]
17	NOT USED	[not currently used]
18	NOT USED	[not currently used]
19	NOT USED	[not currently used]
20	NOT USED	[not currently used]
21	CC_CODE_PREPAYMENT_FOP	Order Configuration - Account - Prepayment FOP
22	CC_CODE REVIEW PHOTO EMAIL SENDER NAME	Pass/Membership - Photo Review Config - E-mail sender name
23	CC_CODE REVIEW PHOTO EMAIL SENDER ADDRESS	Pass/Membership - Photo Review Config - E-mail sender address
24	CC_CODE REVIEW PHOTO EMAIL REPLY ADDRESS	Pass/Membership - Photo Review Config - Reply to E-mail address
25	CC_CODE REVIEW PHOTO REJECTED TEMPLATE ID	Pass/Membership - Photo Review Config - Picture rejected template
26	CC_CODE REVIEW PHOTO VERIFIED USED TEMPLATE ID	Pass/Membership - Photo Review Config - Use current picture template
27	CC_CODE COMPANY NAME	Company - Name
28	CC_CODE COMPANY ABBR	Company - Abbreviation
29	CC_CODE COMPANY NUMBER	Company - ID
30	CC_CODE DEPOSIT LEVEL	Amounts - Deposit Level
31	CC_CODE CHECKS DEFAULT FOP	Authorization - Default FOP for swiped checks

32	CC_CODE_MICR_FORMAT	Authorization - MICR Device Format
33	CC_CODE_USE_CVN	Authorization - Enable Card Verification Number
34	CC_CODE_USE_AVIS	Authorization - Enable address Verification Service
35	CC_CODE_AVIS_ZIP_ONLY	Authorization - Verification Method
36	CC_CODE_ENABLE_MULTIPLE_AUTH	Authorization - Enable multiple authorizations per transaction
37	CC_CODE_ALLOW_OFFLINE_CHECKS	Authorization - Allow offline transactions
38	CC_CODE_PREVENT_DEBIT_FAILURE_RECEIPTS	Authorization - option is true when Print receipts for failed debit authorization attempts is NOT selected
39	CC_CODE_JOURNALIZE_CODE_45_SETTLEMENTS	Authorization - Journalize code 45 settlements for authorizations (not recommended)
40	CC_CODE_PREVENT_OFFLINE_TRANSACTION RETURNS	Authorization - Prevent the return of tickets that were sold in offline transactions
41	CC_CODE_ENCRYPT_AUTH	Authorization - Encrypt sensitive data communicated with Payment Server
42	CC_CODE_AGENCY_COMPOSITE_INCOME	Agency Reports - Composite Income
43	CC_CODE_AGENCY_COMPANY_INCOME	Agency Reports - Company Income
44	CC_CODE_AGENCY_COMMISION	Agency Reports - Commission
45	CC_CODE_AGENCY_SERIAL_NUMBERS	Agency Reports - Serial Numbers
46	CC_CODE_AGENCY_TICKET_DETAIL	Agency Reports - Ticket Detail
47	CC_CODE_AGENCY_SALES_BY_PRODUCT	Agency Reports - Sales Dollars by Product
48	CC_CODE_AGENCY_SALES_QTY_BY_PRODUCT	Agency Reports - Sales Quantity by Product
49	CC_CODE_AGENCY_SALES_BY_HOUR	Agency Reports - Sales by Hour
50	CC_CODE_AGENCY_ADMISIONS	Agency Reports - Admissions
51	CC_CODE_AGENCY_DRAFT_AUDIT_DETAIL	Agency Reports - Draft Audit Detail
52	CC_CODE_AGENCY_DRAFT_AUDIT_TOTALS	Agency Reports - Draft Audit Total
53	CC_CODE_AGENCY_VOID_DETAIL	Agency Reports - Void Detail
54	CC_CODE_AGENCY_SALES_BY_PERSON	Agency Reports - Sales By Salesperson
55	CC_CODE_AGENCY_SALES_BY_TYPE	Agency Reports - Sales by Type
56	CC_CODE_AGENCY_SALES_BY_PROMO	Agency Reports - Sales by Promotion
57	CC_CODE_AGENCY_DISCOUNTS_BY_PRODUCT	Agency Reports - Discounts By Product
58	CC_CODE_AGENCY_COMPANY_INCOME_BY_FOP	Agency Reports - Company Income by FOP
59	CC_CODE_AGENCY_INCOME_BY_PRICE	Agency Reports - Income by Price
60	CC_CODE_AGENCY_TIME_CARD	Agency Reports - Time Card
61	CC_CODE_AGENCY_RETAIL_ITEM_SALES	Agency Reports - Retail Item Sales
62	CC_CODE_AGENCY_DEFERRED_REVENUE	Agency Reports - Deferred Revenue
63	CC_CODE_AGENCY_OVER_SHORT	Agency Reports - Over/Short
64	CC_CODE_AGENCY_OVER_SHORT_SUMMARY	Agency Reports - Over/Short Summary
65	CC_CODE_AGENCY_CUSTOMER_PAYMENTS	Agency Reports - Customer Payments
66	CC_CODE_AGENCY_INCOME_BY_ADMISIONS	Agency Reports - Admissions
67	CC_CODE_AGENCY_VARIANCE	Agency Reports - Variance
68	CC_CODE_AGENCY_ITEM_DISCOUNTS	Agency Reports - Item Discounts
69	CC_CODE_AGENCY_EVENT_SALES	Agency Reports - Event Sales
70	CC_CODE_AGENCY_ITEM_SALES_SUMMARY	Agency Reports - Item Sales Summary
71	CC_CODE_AGENCY_PROJECTIONS	Agency Reports - Projections
72	CC_CODE_AGENCY_MEMBER_USAGE_REPORT	Agency Reports - Member Usage
73	CC_CODE_AGENCY_ITEM_SALES_BY_COA	Agency Reports - Item Sales By COA
74	CC_CODE_AGENCY_SYSTEM_ADJUSTMENT_DETAIL	Agency Reports - System Adjustment Detail
75	CC_CODE_AGENCY_MONTHLY_SALES_BY_NODE	Agency Reports - Monthly Net Sales by Node
76	CC_CODE_AGENCY_SALES_BY_PAYMENT_AND_TYPE	Agency Reports - Sales by Payment and Type
77	CC_CODE_AGENCY_PASS_REQ_TICKET_USAGE	Agency Reports - Pass-Required Ticket Usage
78	CC_CODE_AGENCY_SALES_TYPE_BY_HOUR	Agency Reports - Sales Type by Hour
79	CC_CODE_AGENCY_ORDER_COMPANY_INCOME_BY_FOP	Agency Reports - Order Company Income by FOP
80	CC_CODE_AGENCY_SALES_BY_ITEM_GROUP	Agency Reports - Sales by Item Group
81	CC_CODE_AGENCY_COMPOSITE_INCOME_BY_TICKET_DATE	Agency Reports - Composite Income by Ticket Date
82	CC_CODE_AGENCY_COMPOSITE_INCOME_WITH_TAX	Agency Reports - Composite Income with Tax
83	CC_CODE_AGENCY_STOCK	Agency Reports - Stock
84	CC_CODE_AGENCY_SALES_SUMMARY_BY_TIME	Agency Reports - Sales Summary by Time
85	CC_CODE_AGENCY_PAYMENTS_BY_USER	Agency Reports - Payments by User
86	CC_CODE_AGENCY_PAYMENTS_BY_USER_SUMMARY	Agency Reports - Payments by User Summary
87	CC_CODE_AGENCY_PROMOTIONS_BY_ITEM	Agency Reports - Promotions by Item
88	CC_CODE_AGENCY_PROMOTIONS_BY_ITEM_SUMMARY	Agency Reports - Promotions by Item Summary
89	CC_CODE_AGENCY_VOID_TRANSACTIONS	Agency Reports - Void Transactions
90	CC_CODE_AGENCY_VOID_TRANSACTIONS_SUMMARY	Agency Reports - Void Transactions Summary
91	CC_CODE_AGENCY_SALES_BY_ITEM_GROUP_DETAIL	Agency Reports - Sales by Item Group - Detail
92	CC_CODE_AGENCY_DETAIL_SALES_BY_EVENT	Agency Reports - Detail Sales by Event
93	CC_CODE_AGENCY_TRANSPORTATION_COMMISSIONS	Agency Reports - Transportation Commissions
94	CC_CODE_DRAWER_ASSIGNMENT	Cash Drawers - Keep track of cash drawer assignments
95	CC_CODE_NO_SALES_WHILE_DRAWER_OPEN	Cash Drawers - Prevent sales while drawer is open
96	CC_CODE_CASH_DRAWER1_PNO	Cash Drawers - Cash Drawer 1 - Printer
97	CC_CODE_DRAWER1_COMMAND_NO	Cash Drawers - Cash Drawer 1 - Command
98	CC_CODE_CASH_DRAWER2_PNO	Cash Drawers - Cash Drawer 2 - Printer
99	CC_CODE_DRAWER2_COMMAND_NO	Cash Drawers - Cash Drawer 2 - Command

100	CC_CODE_CASHBACK_PROMPT	Cashback - Prompt
101	CC_CODE_MAX_CASHBACK	Cashback - Maximum
102	CC_CODE_BASE_CURRENCY_NAME	Currencies - Name
103	CC_CODE_BASE_CURRENCY_ABBR	Currencies - Abbreviation
104	CC_CODE_SHOW_DOUBLE_CURRENCY	Currencies - Show Double Currency
105	CC_CODE_DISPLAY_DELAY	Customer Display - Delay before banner in seconds
106	CC_CODE_DISPLAY_BANNER	Customer Display - Banner
107	CC_CODE_DISPLAY_MODE	Customer Display - Mode
108	CC_CODE_DISPLAY_BANNER_TYPE	Customer Display - Banner Type
109	CC_CODE_DISPLAY_CUST_NAME	Customer Display - Show Customer Name
110	CC_CODE_DISPLAY_SUPPRESS_CUST	Customer Display - option is True when Show Customer Amount is NOT selected
111	CC_CODE_DISPLAY_DISCOUNT_AMT	Customer Display - Show Discount Amount
112	CC_CODE_SHOW_TRANSACTION_NO_ON_DISPLAY	Customer Display - Show Transaction Number
113	CC_CODE_CUSTOMER_LOOKUP_FLAG	Customers - Customer Lookup
114	CC_CODE_SHARED_CUST_FILE	Customers - Shared File - disabled if customers data source is SQL
115	CC_CODE_CUSTOMER_FILE_PATH	Customers - Shared File Path - disabled if customers data source is SQL or Shared File not selected
116	CC_CODE_SUPPRESS_CUST_SAVE_MODE	Customers - Save Mode - disabled if customers data source is SQL
117	CC_CODE_CUST_CAT_FOP	Customers - Use Customer Category Forms of Payment instead of the Customer's - disabled if customers data source is SQL
118	CC_CODE_NUM_CUST_ACCOUNT	Customers - Numeric Accounts
119	CC_CODE_ALLOW_BLANK_EXTERNAL_ACCOUNT	Customers - option is true when Use Customer External Account is NOT selected
120	CC_CODE_SHOW_CUST_ADDRESS	Customers - Show Customer Address in Customer Search Results - disabled if customers data source is SQL
121	CC_CODE_ENFORCE_UNIQUE_EXTERNAL_ACCOUNT	Customers - Enforce Unique Customer External Account
122	CC_CODE_MIN_CUST_BROWSER_CAT	Customers - Minimum Category- disabled if customers data source is SQL
123	CC_CODE_MIN_CUST_BROWSER_SUBCAT	Customers - Minimum Subcategory- disabled if customers data source is SQL
124	CC_CODE_MAX_CUST_BROWSER_CAT	Customers - Maximum Category- disabled if customers data source is SQL
125	CC_CODE_MAX_CUST_BROWSER_SUBCAT	Customers - Maximum Subcategory- disabled if customers data source is SQL
126	CC_CODE_CONTINUOUS_ITEMS	Items - Continuous Item Selection
127	CC_CODE_FREEFORM_ENTRY	Items - Allow Free Form entry
128	CC_CODE_RECORD_ITEM_CANCELLATION	Items - Record Cancellation
129	CC_CODE_COMPOUND_ITEM_DISCOUNTS	Items - Compound Item Discounts
130	CC_CODE_DISABLE_PRODUCTS	Items - Disable Product Editing
131	CC_CODE_REMOVE_PLU.LEADING_ZEROS	Items - Remove Leading Zeros from the PLU
132	CC_CODE_REQUIRE_ITEM_VARIANCE_ACCOUNT	Items - Require Item Variance Account.
133	CC_CODE_LOG_NETWORK_ERRORS	Network - Log Network Errors
134	CC_CODE_TRANSFER_LOG	Network - Log Network Activity
135	CC_CODE_MODEM_DELAY	Network - Modem Delay
136	CC_CODE_PICK_NAME_SRC	Picklist - Source for product picklist names
137	CC_CODE_DECIMAL_SEPARATOR	Regional Settings - Decimal Symbol
138	CC_CODE_THOUSAND_SEPARATOR	Regional Settings - Digit grouping symbol
139	CC_CODE_CURRENCY_PLACES	Regional Settings - number of digits after decimal
140	CC_CODE_TWELVE_HR_TIME_FORMAT	Regional Settings - 12 hour time format
141	CC_CODE_DATE_FORMAT	Regional Settings - Short Date Style
142	CC_CODE_DATE_SEPERATOR	Regional Settings - Date separator
143	CC_CODE_EPOCH_YEAR	Regional Settings - Earliest 20 <sup>th</sup> Century year
144	CC_CODE_DDMMYY_DATE	Regional Settings - option value depends on Short Date Style
145	CC_CODE_USE FOUR DIGIT YEARS	Regional Settings - option value depends on Short Date Style
146	CC_CODE_ROUND_COMPANY	Rounding - Base Currency - Company
147	CC_CODE_ROUND_CATEGORY	Rounding - Base Currency - Category
148	CC_CODE_ROUND_SUB_CATEGORY	Rounding - Base Currency - Subcategory
149	CC_CODE_ROUND_DIRECTION	Rounding - Base Currency - Round Direction
150	CC_CODE_ROUND_DENOMINATION	Rounding - Base Currency - Round to Denomination
151	CC_CODE_FOREIGN_ROUND_DIRECTION	Rounding - Foreign Currency - Round Direction
152	CC_CODE_FOREIGN_ROUND_DENOMINATION	Rounding - Foreign Currency - Round to Denomination
153	CC_CODE_CCF_FILE_DEFAULT_DIRECTORY	CCF - Default CCF File Directory
154	CC_CODE_CCF_RETired_FILE_DIRECTORY	CCF - Retired CCF File Directory
155	CC_CODE_DEFAULT_TICKET_DATE	Date Specific Tickets - Use Default Ticket Date
156	CC_CODE_PAST_TICKET_DATES	Date Specific Tickets - Allow Past Ticket Dates
157	CC_CODE_ALLOW_BLOCKOUT_DATES	Date Specific Tickets - Allow a Blocked-Out Date for Ticket Date
158	CC_CODE_RESERVATION_DISPLAY_MODE	Date Specific Tickets - Capacity - Display
159	CC_CODE_JF_SUPPRESS_ALL	Journal - General - Suppress All
160	CC_CODE_JF_CASHOUT_DASH	Journal - General - Dash Cashouts
161	CC_CODE_ADDITIONAL_RECORDS	Journal - General - Additional Records
162	CC_CODE_ECHO_RECEIPT_JOURNAL	Journal - General - Echo Journal on receipt printer
163	CC_CODE_JOURNALIZE_NO_SALE	Journal - General - Journalize "No Sale" transactions
164	CC_CODE_JF_TICKET_SUMMARY	Journal - General - Ticket Summary
165	CC_CODE_ADJUSTMENT_DAYS	Journal - General - Adjustment Limit is the number of previous days in which adjustments may be made
166	CC_CODE_ENCRYPTION_TIMER	Journal - General - Encryption View Timer in seconds

167	CC_CODE_REQUIRE_ENCRYPT_VIEW_SQL_TRACKING	Journal - General - Central Database Tracking
168	CC_CODE_VIEWABLE_JNL_TRANSACTIONS	Journal - General - Limit Journal View
169	CC_CODE_AUTO_DELETE_CC	Journal - General - Delete Credit Card Numbers
170	CC_CODE_RECORD_AGE	Journal - General - Age of Record (In days)
171	CC_CODE_MASK_CHAR	Journal - General - Replacement Character
172	CC_CODE_USE_SYSTEM_COMPANY	Journal - General - Record To Company
173	CC_CODE_JOURNAL_BACKUP	Journal - Backup - Auto backup journal
174	CC_CODE_JOURNAL_BACKUP_PATH	Journal - Backup - Backup Path
175	CC_CODE_JOURNAL_BACKUP_SIZE	Journal - Backup - Megabytes
176	CC_CODE_CAPTURE_POS_SCREEN_SIGNATURE	Signature Capture - Capture signature on POS screen
177	CC_CODE_SIGNATURE_ORIENTATION	Signature Capture - Signature box orientation
178	CC_CODE_SIGNATURE_POSITION	Signature Capture - Signature capture position
179	CC_CODE_DELETE_SIGNATURE_FILES	Signature Capture - Delete local signature files after trans
180	CC_CODE_HIDE_CURSOR_OVER_SIGNATURE	Signature Capture - Hide cursor when signing signature
181	CC_CODE_SC_APPLICATION_ID	Smart Card - Galaxy Level - Application ID
182	CC_CODE_SC_COMPANY_ID	Smart Card - Galaxy Level - Company ID
183	CC_CODE_SC_TDF_FILE	Smart Card - Galaxy Level - TDF File
184	CC_CODE_SC_TDF_MASK_INDEX	Smart Card - Galaxy Level - TDF Mask Index
185	CC_CODE_SC_HS_APPLICATION_ID	Smart Card - Handshake Level - Application ID
186	CC_CODE_SC_HS_COMPANY_ID	Smart Card - Handshake Level - Company ID
187	CC_CODE_SC_HS_TDF_FILE	Smart Card - Handshake Level - TDF File
188	CC_CODE_SC_HS_TDF_MASK_INDEX	Smart Card - Handshake Level - TDF Mask Index
189	CC_CODE_SC_LFS_APPLICATION_ID	Smart Card - Locker Freespace - Application ID
190	CC_CODE_SC_LFS_COMPANY_ID	Smart Card - Locker Freespace - Company ID
191	CC_CODE_SC_LFS_TDF_FILE	Smart Card - Locker Freespace - TDF File
192	CC_CODE_SC_LFS_TDF_MASK_INDEX	Smart Card - Locker Freespace - TDF Mask Index
193	CC_CODE_SC_LFS_VENDOR_ID	Smart Card - Locker Freespace - Vendor ID
194	CC_CODE_SC_LFS_SEGMENT_TYPE	Smart Card - Locker Freespace - Segment Type
195	CC_CODE_STOCK_COMPANY	Stock Tracking - Stock Company
196	CC_CODE_STOCK_CATEGORY	Stock Tracking - Stock Category
197	CC_CODE_STOCK_SUBCAT	Stock Tracking - Stock Sub Category
198	CC_CODE_STOCK_FKEY	Stock Tracking - Stock Fkey
199	CC_CODE_STOCK_PROD_NUM	Stock Tracking - Product Number
200	CC_CODE_STOCK_TYPE	Stock Tracking - Stock Type
201	CC_CODE_PRODUCT_NAME	Titles - Product Name
202	CC_CODE_IDLE_SCREEN_DISPLAY	Titles - Logon
203	CC_CODE_FACTOR_TITLE	Titles - Factor
204	CC_CODE_HIDE_TKT_LK_UP_TICKET	Ticket Lookup - option is true when Display Ticket is NOT selected
205	CC_CODE_HIDE_TKT_LK_UP_TICKET_VISUAL_ID	Ticket Lookup - option is true when Visual ID is NOT selected
206	CC_CODE_HIDE_TKT_LK_UP_TICKET_TICKET_DATE	Ticket Lookup - option is true when Ticket Date is NOT selected
207	CC_CODE_HIDE_TKT_LK_UP_TICKET_EXPIRATION	Ticket Lookup - option is true when Expiration is NOT selected
208	CC_CODE_HIDE_TKT_LK_UP_TICKET_STATUS	Ticket Lookup - option is true when Status is NOT selected
209	CC_CODE_HIDE_TKT_LK_UP_TICKET_PLU	Ticket Lookup - option is true when PLU is NOT selected
210	CC_CODE_HIDE_TKT_LK_UP_TICKET_PLU_DESC	Ticket Lookup - option is true when PLU Description is NOT selected
211	CC_CODE_HIDE_TKT_LK_UP_TICKET_EVENT_NAME	Ticket Lookup - option is true when Event Name is NOT selected
212	CC_CODE_HIDE_TKT_LK_UP_TICKET_EVENT_START	Ticket Lookup - option is true when Event Start Date is NOT selected
213	CC_CODE_HIDE_TKT_LK_UP_TICKET_ORDER_ID	Ticket Lookup - option is true when Order ID is NOT selected
214	CC_CODE_HIDE_TKT_LK_UP_TICKET_PASS	Ticket Lookup - option is true when Pass is NOT selected
215	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT	Ticket Lookup - option is true when Display Accounting is NOT selected
216	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_PRICE	Ticket Lookup - option is true when Price is NOT selected
217	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_TAX	Ticket Lookup - option is true when Tax is NOT selected
218	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_REMAIN_PRICE	Ticket Lookup - option is true when Remaining Price is NOT selected
219	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_REMAIN_TAX	Ticket Lookup - option is true when Remaining Tax is NOT selected
220	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_EXCHANGE	Ticket Lookup - option is true when Exchangeable is NOT selected
221	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_TAX_METHODS	Ticket Lookup - option is true when Taxes is NOT selected
222	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_UPGRADE	Ticket Lookup - option is true when Upgrade Value is NOT selected
223	CC_CODE_HIDE_TKT_LK_UP_ACCESS	Ticket Lookup - option is true when Display Access is NOT selected
224	CC_CODE_HIDE_TKT_LK_UP_ACCESS_CODE	Ticket Lookup - option is true when Access Code is NOT selected
225	CC_CODE_HIDE_TKT_LK_UP_ACCESS_NAME	Ticket Lookup - option is true when Access Name is NOT selected
226	CC_CODE_HIDE_TKT_LK_UP_ACCESS_STATUS	Ticket Lookup - option is true when Update Status is NOT selected
227	CC_CODE_HIDE_TKT_LK_UP_ACCESS_REMAIN	Ticket Lookup - option is true when Remaining Use is NOT selected
228	CC_CODE_HIDE_TKT_LK_UP_ACCESS_USE	Ticket Lookup - option is true when Use Count is NOT selected p -
229	CC_CODE_HIDE_TKT_LK_UP_ACCESS_LOCKEDOUT	Ticket Lookup - option is true when Locked Out is NOT selected
230	CC_CODE_HIDE_TKT_LK_UP_ACCESS_REASON	Ticket Lookup - option is true when Reason is NOT selected
231	CC_CODE_HIDE_TKT_LK_UP_TRANSACTION	Ticket Lookup - option is true when Display Transaction is NOT selected
232	CC_CODE_HIDE_TKT_LK_UP_TRANSACTION_DATE	Ticket Lookup - option is true when Date of Sale is NOT selected
233	CC_CODE_HIDE_TKT_LK_UP_TRANSACTION_NUMBER	Ticket Lookup - option is true when Transaction Number is NOT selected
234	CC_CODE_HIDE_TKT_LK_UP_TRANSACTION_POS_NODE	Ticket Lookup - option is true when POS Node Number is NOT selected
235	CC_CODE_HIDE_TKT_LK_UP_TRANSACTION_VIEW	Ticket Lookup - option is true when View Transaction is NOT selected

236	CC_CODE_TICKET_LOOKUP_DISPLAY_COMPLETE_REFILL_USAGE	Ticket Lookup - Display usage for complete replenish ticket chain when you look up a child ticket
237	CC_CODE_TRANS_PROMPT_ENABLED	Transaction Prompt - Enable Prompt
238	CC_CODE_TRANS_PROMPT_ANS_LEN	Transaction Prompt - Length
239	CC_CODE_TRANS_PROMPT_ID	Transaction Prompt - ID
240	CC_CODE_TRANS_PROMPT_FREQ	Transaction Prompt - Frequency
241	CC_CODE_TRANS_PROMPT_NAME	Transaction Prompt - Name
242	CC_CODE_TRANS_PROMPT	Transaction Prompt - Prompt
243	CC_CODE_SURVEY_ID	Transaction Prompt - Survey ID
244	CC_CODE_TRANS_NOTES_ENABLED	Transaction Prompt - Prompt for a note for every transaction
245	CC_CODE_TRANS_PROMPT_BEFORE_TRANS	Transaction Prompt - When to prompt
246	CC_CODE_VALIDATE_ZIP_ENABLED	Transaction Prompt - Validate for every zip code entered
247	CC_CODE_RESTRICT_ITEMS_BY_PROFILE	Users - General - Restrict items by profile
248	CC_CODE_RESTRICT_USER_CREATION_BY_AGENCY	Users - General - Restrict user creation by agency
249	CC_CODE_LOGON_KIND	Users - Logons - Logon Method
250	CC_CODE_RESTRICT_LOGONS_BY_AGENCY	Users - Logons - Restrict logons by agency
251	NOT USED	[not currently used]
252	CC_CODE_PASS_PURCHASER_KIND	Pass / Membership - Purchaser Kind
253	CC_CODE_EXTRA_PASS_PICKLIST_DATA	Pass / Membership - Extra Picklist Data
254	CC_CODE_AUTO_ASSIGN_PRIMARY	Pass / Membership - Auto-assign Primary
255	CC_CODE_SUPPRESS_JNL_PASS	Pass / Membership - Check to store new or changed pass information in sales journal
256	CC_CODE_SHOW_PASS_LIST	Pass / Membership - Check to display picklist during Pass verification
257	CC_CODE_UNIQUE_PASS_ID_REQ	Pass / Membership - Check if pass ID must be unique
258	CC_CODE_CLEAR_PASS_AFTER_TRANS	Pass / Membership - Check to retain pass information from the last transaction
259	CC_CODE_RESTRICT_PASS_USAGE,	Pass / Membership - Check to restrict usage by pass kind for pass required tickets
260	CC_CODE_UPPERCASE_PASS_DATA	Pass / Membership - Check to use all uppercase letters when editing a pass
261	CC_CODE_PASS_PURCHASER_REQ	Pass / Membership - Check if a pass purchaser must be selected to sell a new pass
262	CC_CODE_SUPPRESS_PASS_REQUIRED_FIELDS	Pass / Membership - Check to not enforce data requirement for the required pass data entry fields
263	CC_CODE_USE_COPY_PIC_LIST	Pass / Membership - Check to always display pass picklist when copying pictures between passes
264	CC_CODE_SHOW_PASS_USAGE	Pass / Membership - Check to show history for invalid pass-required tickets
265	CC_CODE_UPDATE_PURCHASER_KIND	Pass / Membership - Check to update pass purchaser kind based on current config
266	CC_CODE_ALWAYS_COPY_PRIMARY	Pass / Membership - Check to always overwrite existing pass data with the primary pass holder's data
267	CC_CODE_LINK_NON_PASS_REQ	Pass / Membership - Journalize memos linking default pass with tickets and items
268	CC_CODE_ONE_PASS_FOR_MULT_TKTS_PER_TRANS	Pass / Membership - Check to restrict pass-required pass usage to once per transaction
269	CC_CODE_RENEW_EXPIRED_PASS	Pass / Membership - Allow renewal for voided and returned passes
270	[ Replaced by Code 753 ]	Not in use.
271	CC_CODE_VERIFY_PASS_PICTURE	Pass / Membership - When a pass-required item is purchased, the guest's pass photo displays so you can verify their identity
272	CC_CODE_KEEP_PASS_OPEN_DATE_ON_REISSUE	Pass / Membership - When reissuing to a new pass record, the new pass will retain the open date from the old pass
273	CC_CODE_BYPASS_PASS_ACCOUNT_FALLBACK	Pass / Membership - When option is disabled locate a pass by ID if it cannot be found by Visual ID
274	CC_CODE_ALLOW_PASS_LOOKUP	Pass / Membership LookUp
275	CC_CODE_PASS_AGE_RESTRICT	Pass / Membership - Restrict the age of child joint members
276	CC_CODE_PASS_CHILD_MAX_AGE	Pass / Membership - the maximum age that a joint member can be considered a child
277	CC_CODE_EXCLUDE_JOINT_MEMBERS_ON_PASS_SEARCH	Pass / Membership - Exclude joint members by default when searching for passes
278	CC_CODE_PASS_MODULE_THEME	Pass / Membership Module theme
279	CC_CODE_BLIND_BALANCE	POS - General - Blind Balance at Cashout
280	CC_CODE_SUPPRESS_STOCK_USAGE	POS - General - Suppress Stock Usage Calculation
281	CC_CODE_SUPPRESS_DISCOUNT_LIST	POS - General - Suppress Discounts Picklist
282	CC_CODE_CHANGE_FUND_PROMPT	POS - General - Change Fund Prompt at Start of Shift
283	CC_CODE_NO_LEAD_APPROVAL	POS - General - option is True when Lead Open Approval is NOT selected
284	CC_CODE REVIEW CUSTOMER	POS - General - Review Customer when Selected
285	CC_CODE_DISABLE_COMMISSIONS	POS - General - option is True when Compute Commissions is NOT selected
286	CC_CODE_AUTO_CLOSE_SHIFTS	POS - General - Auto Close Shifts
287	CC_CODE_DEFAULT_TRANS_PRODUCT	POS - General - Default Trans Product
288	CC_CODE_TENDERED_REQUIRED	POS - General - Tendered Required
289	CC_CODE_STAY_AT_LEVEL	POS - General - Stay At Level
290	CC_CODE_DISPLAY_DISCOUNT	POS - General - Display Discount
291	CC_CODE_RETURN_VALIDATION	POS - General - Validate Returns
292	CC_CODE_EXCHANGE_AMTS	POS - General - Current Exchange Rate Amounts
293	CC_CODE_ENABLE_DIALER	POS - General - Phone Dialer
294	CC_CODE_DISABLE_ESCAPE_TRANS	POS - General - option is True when Allow Transaction Escaping is NOT selected
295	CC_CODE_MAINTAIN_TAX_ENABLED	POS - General - option is true when Enable Tax After Sale is NOT selected
296	CC_CODE_RECEIPT_WHOLE_PRICES	POS - General - Whole Prices in Receipts
297	CC_CODE_AUTO_EXIT_TENDEX	POS - General - Auto-exit TENDEX
298	CC_CODE_SHOW_ITEM_WITHOUT_TAX	POS - General - option is True when Display Item Prices with Tax is NOT selected
299	CC_CODE_MAIL_ORDER_AUTH	POS - General - Mail order credit only
300	CC_CODE_ACTIVE_DISCOUNTS_ONLY	POS - General - Show Active Discounts
301	CC_CODE_TAX_BY_COMPANY	POS - General - Disbursement Tax by Company
302	CC_CODE_FORCE_MP_MODE_DISPLAY	POS - General - Small Character Totals Display

304	CC_CODE_RETURN_OTHER_TKTS	POS - General - Allow Non-Order Ticket Return
305	CC_CODE_CREATE_RECEIPT_NUMBER	POS - General - Create Receipt Numbers
306	CC_CODE_ALLOW_ABORT_PRINTING	POS - General - Allow Abort of Ticket Printing
307	CC_CODE_PRICE_INCLUDE_TAX	POS - General - Add tax included items in taxables
308	CC_CODE_AUTO_PROCESS_TENDERED	POS - General - Auto-process Amount Tendered
309	CC_CODE_JOURNALIZE_SUMMARY_RECEIPTS	POS - General - Journalize Summary Receipts
310	CC_CODE_JOURNALIZE_DETAIL_RECEIPTS	POS - General - Journalize Detail Receipts
311	CC_CODE_JOURNALIZE_REPRINT_RECEIPT	POS - General - Journalize Reprint Receipt
312	CC_CODE_ENABLE_RESERVATION_STATUS	POS - General - Enable Date Specific Capacity System
313	CC_CODE_UNVALIDATED_VOID	POS - General - Void without Validation
314	CC_CODE_DISB_REUSE_SERIAL	POS - General - Reuse Disbursement Serial Number
315	CC_CODE_VOID_PARTIAL_TRANSACTIONS	POS - General - Void Transaction on Printing Error
316	CC_CODE_RETURN_ISSUED	POS - General - Allow returns only on issued tickets
317	CC_CODE_DO_NOT_PRINT_RECEIPT_FOR_ZERO_AMOUNT_DEPOSITS	POS - General - Do not print deposit receipt for zero amount deposits
318	CC_CODE_ENABLE_ORDER_RECOVERY_AT_STARTUP	POS - General - Enable Order Recovery process at startup
319	CC_CODE REVIEW CUSTOMER WITH STANDARD EDIT	POS - General - Use Customer Edit Screen When Reviewing Customer
320	CC_CODE_PREVENT_NOTE_DISPLAY_FOR_CUSTOMER_TRANSACTIONS	POS - General - Prevent display of customer notes for customer transactions
321	CC_CODE_EXCLUDE_CHANGE_FUND_IN_CASH_DUE_CALCULATION	POS - General - Exclude change fund amount when calculating cash due
322	CC_CODE_SALES_CHANNELUSES_CENTRAL_DB	POS - General - Sales Channel uses Central Database
323	CC_CODE_ONLY_SERIALIZED_TICKETS_FOR_NSales	POS - General - Only include serialized tickets towards @N_SALES cashout keyword
324	CC_CODE_SHOW_MESSAGE_ON_INVALID_SCAN	POS - General - Show Message on Invalid Scan
325	NOT USED	NOT USED
326	CC_CODE_DISABLE_ALT_F4_EXIT_ON_LOGON_SCREEN	POS - General - Disable Alt+F4 exit from the logon screen
327	CC_CODE_SUPPRESS_GIFT_UDF	POS - General - option is true when Collect Gift User Defined Fields for Gift Transactions is NOT selected
328	CC_CODE_PROMPT_FOR_PASS_AFTER_PAY	POS - General - Prompt for new pass demographics after payment
329	CC_CODE_USE_CURRENT_DATE_FOR_TICKET_REPRINT	POS - General - Tickets use current date as Date Sold when reprinted
330	CC_CODE_COMBINE_EXACT_MATCH_POS_ENTRIES	POS - General - Combine Exact Match Entries in POS
331	CC_CODE_CASHOUT_PNO	POS - Ticket Sets - Cashout Report printer number
332	CC_CODE_CASHOUT_TKTSET	POS - Ticket Sets - Cashout Report ticket set
333	CC_CODE_RECEIPT_PNO	POS - Ticket Sets - Reprint Receipt printer number
334	CC_CODE_RECEIPT_TKTSET	POS - Ticket Sets - Reprint Receipt ticket set
335	CC_CODE_TOTALS_PNO	POS - Ticket Sets - Totals Report printer number
336	CC_CODE_TOTALS_TKTSET	POS - Ticket Sets - Totals Report ticket set
337	CC_CODE_DEPOSIT_PNO	POS - Ticket Sets - Deposit Receipt printer number
338	CC_CODE_DEPOSIT_TKTSET	POS - Ticket Sets - Deposit Receipt ticket set
339	CC_CODE_VOID_PNO	POS - Ticket Sets - Void Receipt printer number
340	CC_CODE_VOID_TKTSET	POS - Ticket Sets - Void Receipt ticket set
341	CC_CODE RETURNS_PNO	POS - Ticket Sets - Return Receipt printer number
342	CC_CODE RETURNS_TKTSET	POS - Ticket Sets - Return Receipt ticket set
343	CC_CODE_DEFAULT_TKTSET_PNO	POS - Ticket Sets - Default printer number
344	CC_CODE_DEFAULT_TKTSET	POS - Ticket Sets - Default ticket set
345	CC_CODE_CLOSE_BATCH_PNO	POS - Ticket Sets - Close Batch Report printer number
346	CC_CODE_CLOSE_BATCH_TKTSET	POS - Ticket Sets - Close Batch Report ticket set
347	CC_CODE_CHANGE_FUND_PNO	POS - Ticket Sets - Start of Shift Receipt printer number
348	CC_CODE_CHANGE_FUND_TKTSET	POS - Ticket Sets - Start of Shift Receipt ticket set
349	CC_CODE_SUMMARY_RECEIPT_PNO	POS - Ticket Sets - Summary Receipt printer number
350	CC_CODE_SUMMARY_RECEIPT_TKTSET	POS - Ticket Sets - Summary Receipt ticket set
351	CC_CODE_X_RECEIPT_PNO	POS - Ticket Sets - X Receipt printer number
352	CC_CODE_X_RECEIPT_TKTSET	POS - Ticket Sets - X Receipt ticket set
353	CC_CODE_ALL_CHANGEFUND_PNO	POS - Ticket Sets - Change Fund Receipt printer number
354	CC_CODE_ALL_CHANGEFUND_TKTSET	POS - Ticket Sets - Change Fund Receipt ticket set
355	CC_CODE_SV_TRANS_RECEIPT_PNO	POS - Ticket Sets - Stored Value Transaction Receipt printer number
356	CC_CODE_SV_TRANS_RECEIPT_TKTSET	POS - Ticket Sets - Stored Value Transaction Receipt ticket set
357	CC_CODE_CASH_RESPONSIBILITY	POS - Cashout - Cash Responsibility
358	CC_CODE_DENOMINATION_DEPOSITS	POS - Cashout - Deposit - Method
359	CC_CODE_DEPOSIT_ID_DATATYPE	POS - Cashout - Deposit - Deposit ID Data Type
360	CC_CODE_UNIQUE_DEPOSIT_ID	POS - Cashout - Deposit - Requires the Deposit ID to be unique
361	CC_CODE_DAYS_DEPOSIT_ID_UNIQUE	POS - Cashout - Deposit - number of days deposit ID is unique
362	CC_CODE_BALANCE_ON_CASHOUT	POS - Cashout - Require a balanced deposit to cashout
363	CC_CODE_N_BALANCE_ATTEMPTS	POS - Cashout - number of unsuccessful attempts to balance
364	CC_CODE_REMOTE_CASHOUT	POS - Cashout - Remote Cashout
365	CC_CODE_CASHOUT_ALL_AGENCIES	POS - Cashout - Remote Cashout - All Agencies
366	CC_CODE_PREVENT_ADDITIONAL_CHANGE_FUND	POS - Cashout - Remote Cashout - Prevent Additional Change Fund
367	CC_CODE_SHIFT_RECAP	Shift Reports - Recap
368	CC_CODE_SHIFT_FORMS_OF_PAYMENT	Shift Reports - Forms of Payment
369	CC_CODE_SHIFT_SERIAL_NUMBERS	Shift Reports - Serial Numbers
370	CC_CODE_SHIFT_COMPOSITE_INCOME	Shift Reports - Composite Income
371	CC_CODE_SHIFT_SUPPORTING_DETAIL	Shift Reports - Supporting Detail

372	CC_CODE_SHIFT_SALES_DOLLARS_BY_PRODUCT	Shift Reports - Sales Dollars by Product
373	CC_CODE_SHIFT_SALES_QTY_BY_PRODUCT	Shift Reports - Sales Quantity by Product
374	CC_CODE_SHIFT_TICKET_DETAIL	Shift Reports - Ticket Detail
375	CC_CODE_SHIFT_COMPANY_INCOME	Shift Reports - Company Income
376	CC_CODE_SHIFT_SALES_BY_HOUR	Shift Reports - Sales by Hour
377	CC_CODE_SHIFT_ADMISSESIONS	Shift Reports - Admissions
378	CC_CODE_SHIFT_BOOTH	Shift Reports - Booth
379	CC_CODE_SHIFT_PROMOTIONS	Shift Reports - Promotions
380	CC_CODE_SHIFT_COMPANY_INCOME_BY_FOP	Shift Reports - Company Income by FOP
381	CC_CODE_SHIFT_ORDER_REFERENCE_REPORT	Shift Reports - Order Reference
382	CC_CODE_SHIFT_OVER_SHORT	Shift Reports - Over Short
383	CC_CODE_SHIFT_OVER_SHORT_SUMMARY	Shift Reports - Over Short Summary
384	CC_CODE_SHIFT_SALES_BY_PROMOTION	Shift Reports - Sales by Promotion
385	CC_CODE_SHIFT_RETAIL_ITEM_SALES	Shift Reports - Retail Item Sales
386	CC_CODE_SHIFT_DETAILED_COMPOSITE_INCOME	Shift Reports - Detailed Composite Income
387	CC_CODE_SHIFT_SALES_TYPE_BY_HOUR	Shift Reports - Sales Type by Hour
388	CC_CODE_SHIFT_COMPOSITE_INCOME_WITH_TAX	Shift Reports - Composite Income With Tax
389	CC_CODE_SHIFT_NO_SALE	Shift Reports - No Sale
390	CC_CODE_SHIFT_SALES_BY_ITEM_GROUP	Shift Reports - Sales by Item Group
391	CC_CODE_SHIFT_STOCK	Shift Reports - Stock
392	CC_CODE_SHIFT_DVR_EXCEPTION	Shift Reports - DVR Exception
393	CC_CODE_SHIFT_DETAIL_SALES_BY_ITEM_GROUP	Shift Reports - Detail Sales by Item Group
394	CC_CODE_INACTIVITY_TIMEOUT	POS - Advanced - Inactivity timeout in minutes
395	CC_CODE_CHANGEFUND_BY_FOP	POS - Advanced - Change Fund by Denomination
396	CC_CODE_CHANGE_FUND_FOP	POS - Advanced - Change Fund form of payment
397	CC_CODE_DISABLE_AUTO_VALIDATE_BY_NODE	POS - Advanced - option is true when Enable Auto-validation is NOT selected
398	CC_CODE_ACTION_ON_TICKET_SCAN	POS - Advanced - Action on Ticket Scan
399	CC_CODE_ISSUANCE_MODE	POS - Advanced - Default Issuance Mode
400	CC_CODE_MAINTAIN_ISSUANCE_STATUS	POS - Advanced - Do not return to default issue mode after each transaction
401	CC_CODE_ENFORCE_DEPOSIT_LEVEL	POS - Cash Drawer Balances - Enforce Deposit Level
402	CC_CODE_OVER_DEPOSIT_LEVEL_TRANS_COUNT	POS - Cash Drawer Balances - number of Transactions permitted while deposit level exceeded
403	CC_CODE_INTERIM_DEPOSIT_AMOUNT_CANNOT_EXCEED_SALES	POS - Cash Drawer Balances - Interim deposit amount cannot exceed sales
404	CC_CODE_USED_TICKET_RETURN_TIMEOUT	POS - Returns - Used Ticket return timeout
405	CC_CODE_ENFORCE_MATCHING_RETURN_FOP	POS - Returns - Require that return FOP matches purchase FOP
406	CC_CODE_DISALLOW_MP RETURNS_WHEN_MATCHING_FOPS	POS - Returns - Deny returns for multi-payments
407	CC_CODE_TABLES_INSERT_CUSTOMERS	Audit logging - audit customer inserts
408	CC_CODE_TABLES_INSERT_ORDERS	Audit logging - audit order inserts
409	CC_CODE_TABLES_INSERT_ORDERLINES	Audit logging - audit order line inserts
410	CC_CODE_TABLES_INSERT_ACCESSCODES	Audit logging - audit access code inserts
411	CC_CODE_TABLES_INSERT_EVENTS	Audit logging - audit event inserts
412	CC_CODE_TABLES_INSERT_ITEMS	Audit logging - audit item inserts
413	CC_CODE_TABLES_INSERT_LOCKOUTS	Audit logging - audit lockout inserts
414	CC_CODE_TABLES_INSERT_PASSES	Audit logging - audit pass inserts
415	CC_CODE_TABLES_INSERT_WAITLISTS	Audit logging - audit waitlist inserts
416	CC_CODE_TABLES_UPDATE_CUSTOMERS	Audit logging - audit customer updates
417	CC_CODE_TABLES_UPDATE_ORDERS	Audit logging - audit order updates
418	CC_CODE_TABLES_UPDATE_ORDERLINES	Audit logging - audit order line updates
419	CC_CODE_TABLES_UPDATE_ACCESSCODES	Audit logging - audit access code updates
420	CC_CODE_TABLES_UPDATE_EVENTS	Audit logging - audit event updates
421	CC_CODE_TABLES_UPDATE_ITEMS	Audit logging - audit item updates
422	CC_CODE_TABLES_UPDATE_LOCKOUTS	Audit logging - audit lockout updates
423	CC_CODE_TABLES_UPDATE_PASSES	Audit logging - audit pass updates
424	CC_CODE_TABLES_UPDATE_WAITLISTS	Audit logging - audit waitlists updates
425	CC_CODE_TABLES_DELETE_CUSTOMERS	Audit logging - audit customer deletes
426	CC_CODE_TABLES_DELETE_ORDERS	Audit logging - audit order deletes
427	CC_CODE_TABLES_DELETE_ORDERLINES	Audit logging - audit order line deletes
428	CC_CODE_TABLES_DELETE_ACCESSCODES	Audit logging - audit access code deletes
429	CC_CODE_TABLES_DELETE_EVENTS	Audit logging - audit event deletes
430	CC_CODE_TABLES_DELETE_ITEMS	Audit logging - audit item deletes
431	CC_CODE_TABLES_DELETE_LOCKOUTS	Audit logging - audit lockout deletes
432	CC_CODE_TABLES_DELETE_PASSES	Audit logging - audit pass deletes
433	CC_CODE_TABLES_DELETE_WAITLISTS	Audit logging - audit wait lists deletes
434	CC_CODE_TABLES_CENTRAL_ITEMS	Items are managed in central maintenance
435	CC_CODE_TABLES_CENTRAL_CONFIG	General Configuration options are managed in central maintenance
436	CC_CODE_END_OF_DAY_TIME	Times - End of Day
437	CC_CODE_SHARED_ITEM_FILE	Tables - Items - Shared File
438	CC_CODE_ITEM_FILE_PATH	Tables - Items - Path
439	CC_CODE_DO_NOT_CACHE_ITEMS	Tables - Items - Do not cache Items during start up

440	CC_CODE_SHARED_COA_FILE	Tables - Chart of Accounts - Shared file
441	CC_CODE_COA_FILE_PATH	Tables - Chart of Accounts - Path
442	CC_CODE_DB_KIND	Tables - Data Source for Customer Transactions
443	CC_CODE_GXKEY_DB_SOURCE	Tables - Data Source for Encryption Keys
444	CC_CODE_USER_DB_SOURCE	Tables - Data Source for Users
445	CC_CODE_LOG_AGENCY_CHANGES	Tables - Rrecord Logging - Agencies
446	CC_CODE_LOG_ITEM_GROUP_CHANGES	Tables - Rrecord Logging - Item Groups
447	CC_CODE_LOG_USER_CHANGES	Tables - Rrecord Logging - Users
448	CC_CODE_LOG_ALL_PASSWORD_CHANGES	Tables - Rrecord Logging - User Pwd Changes
449	CC_CODE_LOG_PROFILE_CHANGES	Tables - Record Logging - User Profiles
450	CC_CODE_LOG_MENU_CHANGES	Tables - Record Logging - Galaxy Menus
451	CC_CODE_PROD_COMM_REPORT	Reporting - Commission Report
452	CC_CODE_SALES_COUNTS_BY_HOUR	Reporting - Sales by Hour - Sales Counts by Hour
453	CC_CODE_SALES_BY_HOUR_DAILY	Reporting - Sales by Hour - Separate Report by Day
454	CC_CODE_PARTIAL_SALES REP	Reporting - Salesperson Report - option is true when "All Tickets" is NOT selected
455	CC_CODE_PARTIAL_PROMO REP	Reporting - Promotion Reports - option is true when "All Tickets" is NOT selected
456	CC_CODE_NON_CASH_STR	Reporting - Non-Cash FOP Label
457	CC_CODE_INVOICE_MEMO_ID	Reporting - Customer Revenue Report - Memo Number
458	CC_CODE_INVOICE_REPORT_FOP	Reporting - Customer Revenue Report - Invoice form of payment
459	CC_CODE_CUST_REV_SUPPRESS_ITEMS	Reporting - Customer Revenue Report - option is true when "Include Items" is NOT selected.
460	CC_CODE_DEFAULT_SALES_BY_ITEM_GROUP_REPORT_DEF	Reporting - Sales by Item Group - Default Definition
461	CC_CODE_SUBTRACT_CHANGE_FUND	Reporting - Assign COA to Change Fund
462	CC_CODE_ONLINE_REPORT_PNO	Reporting - Report Printer
463	CC_CODE_BRIEF_ONLINE_INC REP	Reporting - Brief Income Statement
464	CC_CODE_PRINTER_PAPER_NAME	Reporting - Paper Size
465	CC_CODE_COMB_DISB_SALES_HOUR	Reporting - Sales by Hour
466	CC_CODE_COMB_DISB_ADMISSE	Reporting - Admissions
467	CC_CODE_COMB_DISB_SALES_TYPE	Reporting - Sales by Event type
468	CC_CODE_COMB_DISB_DISC_PROD	Reporting - Discount By Product
469	CC_CODE_COMB_DISB_PROMOTIONS	Reporting - Promotions
470	CC_CODE_COMB_DISB_CUST_INV	Reporting - Customer Invoices
471	CC_CODE_COMB_DISB_ZIPCODE	Reporting - Zipcode
472	CC_CODE_COMB_DISB_CUSTREV	Reporting - Customer Revenue
473	CC_CODE_COMB_DISB_CREVCOA	Reporting - Customer Revenue by COA
474	CC_CODE_COMB_DISB_MONTH REP	Reporting - Monthly Net Sales by Node
475	CC_CODE_COMB_DISB_CUST_MEMO	Reporting - Customer Revenue by Memo
476	CC_CODE_COMB_DISB_ITEM_COA	Reporting - Item Sales by COA
477	CC_CODE_COMB_DISB_ITEM_GROUP	Reporting - Sales By Item Group
478	CC_CODE_SUMMARY_BY_CURRENCY	Reporting - Summary By Currency
479	CC_CODE_EXCHANGE_RATE_KEY	Reporting - Exchange Rate
480	CC_CODE_MOTORCYCLE_ACCESS_CODE_GROUP_ID	Reporting - Motorcycle Access Code Group
481	CC_CODE_VALID_USAGE_ACCESS_CODE_GROUP_ID	Reporting - Valid Usage Access Code Group
482	CC_CODE_POS_SALES_CHANNEL_ID	Sales Channel
483	CC_CODE_USE_GANTNER_GAT_6000_ISO_LOCKERS	Lockers - Use Gantner GAT 6000 ISO
484	CC_CODE_SV_LOOKUP_PNO	SV Config - lookup options - Printer
485	CC_CODE_SV_LOOKUP_TKTSET	SV Config - lookup options - Ticket Set
486	CC_CODE_SV_AUTO_PRINT_BALANCE	SV Config - Automatically print ticket set after lookup
487	CC_CODE_UNLOAD_SV_PNO	SV Config - Unload Options - Printer
488	CC_CODE_UNLOAD_SV_TKTSET	SV Config - Unload Options - Ticket Set
489	CC_CODE_MAX_UNLOAD_AMOUNT	SV Config - Maximum unload amount
490	CC_CODE_SELECT_UNLOAD_FOP	SV Config - Select unload FOP from POS
491	CC_CODE_UNLOAD_GC_PLU	SV Config - Unload Item
492	CC_CODE_ALLOW_MANUAL_GC_ENTRY	SV Config - Allow manual card entry for activation and recharge (for supported SV types)
493	CC_CODE_CHARGE_ACCOUNT_FOP	Order Config - Credit form of payment
494	CC_CODE_DEFAULT RETURNS_FOP	Order Config - Default Returns form of payment
495	CC_CODE_SALES_PROGRAM_DISPLAY_MODE	Order Config - Sales Program Display Mode
496	CC_CODE_FORCE_CUST_CRITERIA	Order Config - Enforce Picklist Search Criteria: Customer
497	CC_CODE_FORCE_ORDER_CRITERIA	Order Config - Enforce Picklist Search Criteria: Order
498	CC_CODE_FORCE_CONTACT_CRITERIA	Order Config - Enforce Picklist Search Criteria: Contact
499	CC_CODE_FORCE_INVOICE_CRITERIA	Order Config - Enforce Picklist Search Criteria: Invoice
500	CC_CODE_FORCE_CM_CRITERIA	Order Config - Enforce Picklist Search Criteria: Credit Memo
501	CC_CODE_ISSUE_START_PNO	Order Config - When issuing an order - start ticket set printer
502	CC_CODE_ISSUE_START_TKTSET	Order Config - When issuing an order - start ticket set
503	CC_CODE_ISSUE_END_PNO	Order Config - When issuing an order - end ticket set printer
504	CC_CODE_ISSUE_END_TKTSET	Order Config - When issuing an order - end ticket set
505	CC_CODE_COPY_ORDER_NOTE	Order Config - When copying an order, include Note
506	CC_CODE_COPY_ORDER_OPTIONAL_FIELDS	When copying an order, include Purchase Order
507	CC_CODE_AUTOMATIC RETURNS_FOP	Order Config - Enforce Default Returns FOP when Returning Tickets
508	CC_CODE_DO_CREDIT_CHECK	Order Config - Enforce Credit Limit when Issuing Ticket

509	CC_CODEFORCE_DELIVERY_METHOD_SELECTION	Order Config - Enforce Delivery Method Selection for New Orders
510	CC_CODESALES_PROGRAM_REQUIRED	Order Config - Sales Program Required
511	CC_CODE_SHOW_TICKET_DATE_IN_COLUMN	Order Config - Show Ticket Date in a Separate Column
512	CC_CODE_IGNORE_ORDER_PAYMENTS	Order Config - Ignore Payments when Altering Sales Programs
513	CC_CODE_ALLOW_CURRENCY_FOR_OE_PAYMENT	Order Config - Allow international currency selection for payments
514	CC_CODE_AUTO_ISSUE_FKEY	Order Config - Allow Auto-Issue
515	CC_CODE_NEW_VISUALID_FOR_REPRINTS	Order Config - Create New VisualID When Reprinting Tickets
516	CC_CODE_CONFIRM_ORDER_ISSUE	Order Config - Confirm ticket issue
517	CC_CODE_ALLOW_OE_AND_PASS_CONTACT_INTERACTION	Order Config - Allow Pass contacts to be used in Order Entry and vice versa
518	CC_CODE_USE_DELIVERY_METHOD_FOR_ISSUE	Order Config - Use Delivery Method to determine how tickets are issued
519	CC_CODE_ALLOW_AUTO_ORDER_CLOSE	Order Config - Allow Automatic Order Closing
520	CC_CODE_COPY_CUSTOMER_NOTES	Order Config - Use Customer Notes as Notes for New Orders
521	CC_CODE_USE_FIRST_EVENT_AS_CURRENT	Order Config - Use first event as current when selecting items
522	CC_CODE_ENFORCE_CREDIT_LIMIT_ON_ADD_ITEMS	Order Config - Enforce credit limit when adding items to order
523	CC_CODE_PREVENT_OE_PAYMENT_WITH_CONTRACT	Order Config - Prevent payments if order has contract
524	CC_CODE_FORCE_ORDER_NOTES REVIEW_IN_QOP	Order Config - Force Order Notes Review in Quick Order Pickup
525	CC_CODE_BATCH_START_PNO	Orders - Batch Print - Batch start printer number
526	CC_CODE_BATCH_START_TKTSET	Orders - Batch Print - Batch start ticket set
527	CC_CODE_BATCH_END_PNO	Orders - Batch Print - Batch end printer number
528	CC_CODE_BATCH_END_TKTSET	Orders - Batch Print - Batch end ticket set
529	CC_CODE_ORDER_START_PNO	Orders - Batch Print - Order start printer number
530	CC_CODE_ORDER_START_TKTSET	Orders - Batch Print - Order start ticket set
531	CC_CODE_ORDER_END_PNO,	Orders - Batch Print - Order end printer number
532	CC_CODE_ORDER_END_TKTSET,	Orders - Batch Print - Order end ticket set
533	CC_CODE_BATCH_TIME_INTERVAL,	Orders - Batch Print - time interval in seconds
534	CC_CODE_BATCH_TICKET_PAUSE,	Orders - Batch Print - pause at ticket quantity
535	CC_CODE_BATCH_TICKET_TYPE_PAUSE,	Orders - Batch Print - pause at ticket type
536	CC_CODE_BATCH_PRINT_PAUSE,	Orders - Batch Print - pause at order
537	CC_CODE_NO_SUSPEND_BATCH_PRINT,	Orders - Batch Print - option is true when 'Allow Batch Print to be Suspended' is NOT selected
538	CC_CODE_NO_BATCH_PRINT_ORDER_STATUS,	Orders - Batch Print - option is true when 'Include Open Orders Only' is checked
539	CC_CODE_BATCH_STATEMENT_ISSUED_ONLY,	Orders - Batch Print - option is true when 'Print only issued lines on statement' is selected
540	CC_CODE_SELECT_ENTIRE_ORDER_FOR_BATCH_PRINT	Orders - Batch Print - select entire order for batch print
541	CC_CODE_BATCH_STATEMENT_ID	Orders - Statements - batch template option
542	CC_CODE_INVOICE_STATEMENT_ID	Orders - Statements - invoice template option
543	CC_CODE_REFUND_STATEMENT_ID	Orders - Statements - credit memo template option
544	CC_CODE_PICKUP_STATEMENT_ID	Orders - Statements - pickup template option
545	CC_CODE_COMBINE_ISSUED_ITEMS	Orders - Statements - combine issued items option
546	CC_CODE_PREVIEW_STATEMENT	Orders - Statements - preview statement option
547	CC_CODE_AUTOSAVE_STATEMENT	Orders - Statements - auto save statement option
548	CC_CODE_AUTOSAVE_PATH	Orders - Statements - path to folder for saved statement
549	CC_CODE_NUM_OF_STATEMENTS_TO_PRINT	Orders - Statements - number of copies to print
550	CC_CODE_PICKUP_AUTO_PRINT_REPORT	Orders - Pickup - Auto print report option
551	CC_CODE_PICKUP_PNO	Orders - Pickup - Printer Number for add'l ticket set
552	CC_CODE_PICKUP_TKTSET	Orders - Pickup - Additional Ticket Set
553	CC_CODE_ALLOW_MULTI_PICKUP	Orders - Pickup - allow multi-pickup option is true when 'Edit Orders after Pickup' is selected
554	CC_CODE_PICKUP_PREVENT_ISSUING_ITEMS	Orders - Pickup - prevent issuing items option is true when 'Issue Items' is NOT selected
555	CC_CODE_AUTO_ISSUE	Orders - Pickup - auto issue option is true when 'Confirm Issue before Pickup' is NOT selected
556	CC_CODE_AUTO_PICKUP	Orders - Pickup - auto pickup option is true when 'Confirm Pickup' is NOT selected
557	CC_CODE_IGNORE_GUEST_ARRIVAL_DATE	Orders - Pickup - ignore guest arrival date option
558	CC_CODE_ORDER_IMPORT_FOLDER	Orders - Import - path for imported order files
559	CC_CODE_PROCESSED_ORDER_IMPORT_FOLDER	Orders - Import - path for processed order files
560	CC_CODE_DEFAULT_CUSTOMER_CATEGORY	Orders - Events - default customer category option
561	CC_CODE_DEFAULT_CUSTOMER	Orders - Events - default customer option
562	CC_CODE_OE_SALES_CHANNEL_ID	Orders - Events - Sales Channel option
563	CC_CODE_INVOICE_REF_REQ	Invoices - reference required option
564	CC_CODE_INVOICE_FOP	Balance Adjustment FOP for Invoices
565	CC_CODE_MIN_INVOICE_AMOUNT	Invoices - minimum amount
566	CC_CODE_MIN_INVOICE_FOP	Invoices - minimum FOP
567	CC_CODE_PARTIAL_REFUND	Credit Memos - partial refunds option
568	CC_CODE_JOURNALIZE_OVERPAYMENTS	Credit Memos - journalize overpayments option
569	CC_CODE_REFUND_FOP	Balance Adjustment FOP for credit memos
570	CC_CODE_MIN_REFUND_COMPANY	Minimum Account for credit memos - Company
571	CC_CODE_MIN_REFUND_CATEGORY	Minimum Account for credit memos - Category
572	CC_CODE_MIN_REFUND_SUBCAT	Minimum Account for credit memos - Sub-Category
573	CC_CODE_MIN_REFUND_AMOUNT	Minimum refund amount for credit memos
574	CC_CODE_THRESHOLD_FOP	Minimum Change FOP for credit memos
575	CC_CODE_REFUND_BREAK_COMPANY	Order Config - Invoice and Credit Memo - Break Company
576	CC_CODE_REFUND_BREAK_CATEGORY	Order Config - Invoice and Credit Memo - Break Category
577	CC_CODE_REFUND_BREAK_SUBCAT	Order Config - Invoice and Credit Memo - Break Sub-Category

578	CC_CODE_GROUP_VISITATION_REPORT_GROUP_BY_GX_UDF_ID	Order User-Defined Field used to group report results
579	CC_CODE_GROUP_VISITATION_REPORT_DETAIL_GX_UDF_ID	Order User-Defined Field to be displayed on the Group Visitations Detail Report
580	CC_CODE_TEMP_PDF_FOLDER	Orders - PrintAtHome - path to folder to store PDF's
581	CC_CODE_DELETE_PDF_FILES	Orders - PrintAtHome - delete PDF files after processing
582	CC_CODE_PRINT_AT_HOME_JPEG_IMAGE_COMPRESSION	Orders - PrintAtHome - image compression option
583	CC_CODE_EMAIL_SENDER_NAME	Orders - PrintAtHome - email senders name and address
584	CC_CODE_EMAIL_SENDER_ADDRESS	Orders - PrintAtHome - sender's email address
585	CC_CODE_REPLY_TO_EMAIL_ADDRESS	Orders - PrintAtHome - The 'reply-to' email address to use for e-mail sent by the application
586	CC_CODE_CONFIRMATION_EMAIL_OPTION	Orders - PrintAtHome - confirmation email option
587	CC_CODE_DEFAULT_OE_CUSTOMER_ID	Default customer used when creating new orders
588	CC_CODE_DEFAULT_OE_CUSTOMER_LABEL	Text to display when default customer is selected
589	CC_CODE_DEFAULT_OE_FILTER_ID	Default Order Entry orders filter
590	CC_CODE_TABLES_CENTRAL_ACCESS_CODES	Access Codes are managed in central maintenance
591	CC_CODE_TABLES_CENTRAL_ACCESS_CODE_GROUPS	Access Code Groups are managed in central maintenance
592	CC_CODE_TABLES_CENTRAL_ACP_MODES	ACP Modes are managed in central maintenance
593	CC_CODE_TABLES_CENTRAL_ACPS	Access Control Points are managed in central maintenance
594	CC_CODE_TABLES_CENTRAL_ADMISSION_LISTS	Admission Lists are managed in central maintenance
595	CC_CODE_TABLES_CENTRAL_AGENCIES	Agencies are managed in central maintenance
596	CC_CODE_TABLES_CENTRAL_BANKS	Banks are managed in central maintenance
597	CC_CODE_TABLES_CENTRAL_BLOCKOUT_DATES	Blockout Dates are managed in central maintenance
598	CC_CODE_TABLES_CENTRAL_BLOCKOUT_GROUPS	Blockout Groups are managed in central maintenance
599	CC_CODE_TABLES_CENTRAL_CALENDARS	Calendars are managed in central maintenance
600	CC_CODE_TABLES_CENTRAL_CARD_RANGES	Card Ranges are managed in central maintenance
601	CC_CODE_TABLES_CENTRAL_COAS	Charts of Account are managed in central maintenance
602	CC_CODE_TABLES_CENTRAL_COMPANIES	Companies are managed in central maintenance
603	CC_CODE_TABLES_CENTRAL_COUNTRIES	Countries are managed in central maintenance
604	CC_CODE_TABLES_CENTRAL_COUPONS	Coupons are managed in central maintenance
605	CC_CODE_TABLES_CENTRAL_DENOMINATIONS	Denominatoins are managed in central maintenance
606	CC_CODE_TABLES_CENTRAL_DISBURSEMENTS	Disbursements are managed in central maintenance
607	CC_CODE_TABLES_CENTRAL_DISCOUNTS	Discounts are managed in central maintenance
608	CC_CODE_TABLES_CENTRAL_DMA	Demographic marketing areas are managed in central maintenance
609	CC_CODE_TABLES_CENTRAL_EXCHANGE_METHODS	Exchange Methods are managed in central maintenance
610	CC_CODE_TABLES_CENTRAL_EXCHANGE_RATES	Exchange Rates are managed in central maintenance
611	CC_CODE_TABLES_CENTRAL_EXTERNAL_CALLS	External Calls are managed in central maintenance
612	CC_CODE_TABLES_CENTRAL_FACILITIES	Facilities are managed in central maintenance
613	CC_CODE_TABLES_CENTRAL_FKEY_TASKS	Fkey Tasks are managed in central maintenance
614	CC_CODE_TABLES_CENTRAL_FOPS	Forms of Payment are managed in central maintenance
615	CC_CODE_TABLES_CENTRAL_FORMATTED_PROMPTS	Formatted prompts are managed in central maintenance
616	CC_CODE_TABLES_CENTRAL_HOLIDAYS	Holidays are managed in central maintenance
617	CC_CODE_TABLES_CENTRAL_ITEM_GROUPS	Item Groups are managed in central maintenance
618	CC_CODE_TABLES_CENTRAL_ITEM_GROUP_REPORT_DEFINITIONS	Item Group Report Definitions are managed in central maintenance
619	CC_CODE_TABLES_CENTRAL_LEAD_PAGE_MENU	Lead Page Menus are managed in central maintenance
620	CC_CODE_TABLES_CENTRAL_LOCKOUTS	Lock Outs are managed in central maintenance
621	CC_CODE_TABLES_CENTRAL_MULTIPLE_CHOICES	Multiple Choices are managed in central maintenance
622	CC_CODE_TABLES_CENTRAL_NODES	Nodes are managed in central maintenance
623	CC_CODE_TABLES_CENTRAL_ONLINE_MENUS	Online Menus are managed in central maintenance
624	CC_CODE_TABLES_CENTRAL_OPERATIONS	Operations are managed in central maintenance
625	CC_CODE_TABLES_CENTRAL_POS_MENUS	POS Menus are managed in central maintenance
626	CC_CODE_TABLES_CENTRAL_POS_MESSAGES	POS Messages are managed in central maintenance
627	CC_CODE_TABLES_CENTRAL_PROJECTIONS	Projections are managed in central maintenance
628	CC_CODE_TABLES_CENTRAL_PROTOCOLS	Protocols are managed in central maintenance
629	CC_CODE_TABLES_CENTRAL_REASONS	Reasons are managed in central maintenance
630	CC_CODE_TABLES_CENTRAL_TAX_TABLES	[Not currently in use]
631	CC_CODE_TABLES_CENTRAL_TERMINAL_IDS	Terminal IDs are managed in central maintenance
632	CC_CODE_TABLES_CENTRAL_TICKET_SETS	Ticket Sets are managed in central maintenance
633	CC_CODE_TABLES_CENTRAL_TICKET_STOCK	Tickt Stock is managed in central maintenance
634	CC_CODE_TABLES_CENTRAL_USER_DEFINED_KEYWORDS	User Defined Keywords are managed in central maintenance
635	CC_CODE_TABLES_CENTRAL_USERS	Users are managed in central maintenance
636	CC_CODE_TABLES_CENTRAL_USER_PROFILES	User profiles are managed in central maintenance
637	CC_CODE_TABLES_CENTRAL_WARNINGS	Warnings are managed in central maintenance
638	CC_CODE_TABLES_CENTRAL_ZIPCODES	Zip Codes are managed in central maintenance
639	CC_CODE_USE_DEFAULT_MENU_BY_DAY	Use default menu by day
640	CC_CODE_USE_PHOTO_STATUS_ON_PASS	Use photo status on pass
641	CC_CODE_MANIFEST_REPORT_EVENT_TYPE_ID	Manifest report event type ID
642	CC_CODE_MANIFEST_REPORT_FOP_SET_ID	Manifest report FOP set ID
643	CC_CODE_MANIFEST_REPORT_DO_NOT_INCLUDE_ZERO_AMOUNT_TICKETS	Manifest report do not include zero amount tickets
644	CC_CODE_TABLES_CENTRAL_PRICE_PROGRAMS	Price Programs are managed in central maintenance
645	CC_CODE_TABLES_CENTRAL_PRICE_SCHEDULES	Price Schedules are managed in central maintenance
646	CC_CODE_TABLES_CENTRAL_PRICE_PROGRAM_TIME_RANGES	Price Program Time Ranges are managed in central maintenance

647	CC_CODE_TABLES_CENTRAL_PRICE_CALENDARS	Price Calendars are managed in central maintenance
648	CC_CODE_KIOSK_SALES_CHANNEL	Kiosk sales channel
649	CC_CODE_KIOSK_CATEGORY_GROUP	Kiosk category group
650	CC_CODE_KIOSK_FOLDER	Kiosk folder - path to theme folder
651	CC_CODE_KIOSK_THEME	Kiosk theme name
652	CC_CODE_KIOSK_DISPLAY_FEES	Kiosk display item and transaction fees
653	CC_CODE_ALLOW_IDLE_KEYPRESS	Kiosk allow idle keypress - screen touch starts a transaction
654	CC_CODE_KIOSK_ALLOW_SWIPE_CONFIRMATION	Kiosk allow swipe at purchase confirmation step
655	CC_CODE_KIOSK_PURCHASE_REQ_CC	Kiosk credit card required to start purchase
656	CC_CODE_KIOSK_TIMEOUT	Kiosk timeout in seconds
657	CC_CODE_KIOSK_END_TRANSACTION_TIMEOUT	Kiosk end transaction timeout
658	CC_CODE_KIOSK_BALANCING_FOP	Kiosk balancing FOP
659	CC_CODE_KIOSK_CREDIT_CARD_RETRIES	Kiosk credit card retries
660	CC_CODE_KIOSK_DO_NOT_DISPLAY SOLD_OUT_EVENTS	Kiosk do not display sold out events
661	CC_CODE_KIOSK_ONLY_SHOW_TODAYS_EVENTS	Kiosk only show today's events
662	CC_CODE_KIOSK_ADVANCED_LOGGING	Kiosk advanced logging
663	CC_CODE_MAX_COUPONS	Maximun number of configurable coupons
664	CC_CODE_MAX_TKTSETS	Maximum number of configurable ticket sets
665	CC_CODE_CLEAR_EVENT_AFTER_TRANS	Clear event after transaction
666	CC_CODE_FORCE_EVENT_SELECTION	Force event selection
667	CC_CODE_ALLOW_DEACTIVATE_SOLD_EVENTS	Allow deactivate sold events
668	CC_CODE_HIDE_ALL_OFF_SALE_EVENTS	Hide all off sale events
669	CC_CODE_DEFAULT_SEARCH_EVENT_AVAIL	Default search criteria for event availability
670	CC_CODE_WAIT_LIST_RESPONSE_CODE_TABLE_ID	Wait list response code table ID
671	CC_CODE_WAIT_LIST_ACTION_CODE_TABLE_ID	Wait list action code table ID
672	CC_CODE_ROUND_TAXES_DOWN	Round taxes down
673	CC_CODE_PTO_ADVANCE_LIMIT	PTO Advance Limit
674	CC_CODE_TRANS_REVERSE_PRINT	Transportation reverse print
675	CC_CODE_DO_NOT_USE_120_RULE	Do not use 120% rule for mileage based fares
676	CC_CODE_TRAN_SUMMARY_PNO	Transportation summary printer number
677	CC_CODE_TRAN_SUMMARY_TKTSET	Transportation summary ticket set
678	CC_CODE_TRAN_DETAIL_PNO	Transportation detail printer number
679	CC_CODE_TRAN_DETAIL_TKTSET	Transportation detail ticket set
680	CC_CODE_REISSUE_ORIGIN	Print reissue from origin
681	CC_CODE_LAYOVER_LIMIT	Layover limit
682	CC_CODE_LAYOVER_MESSAGE	Layover message
683	CC_CODE_LAYOVER_WAIVER_COUPON	Layover waiver coupon
684	CC_CODE_KIOSK_SHUTDOWN_ON_BCAM_NO_RESPONSE	Kiosk shutdown on BCAM no response
685	CC_CODE_TABLES_CENTRAL_TAXES	Taxes are managed in central maintenance
686	CC_CODE_TABLES_CENTRAL_MEDIA_DEFS	Media Definitions are managed in central maintenance
687	CC_CODE_ENFORCE_ITEM_PICKLIST_CRITERIA	Enforce Item picklist criteria
688	CC_CODE_TABLES_CENTRAL_SURVEYS	Surveys are managed in central maintenance
689	CC_CODE REVIEW CUSTOMER EMAIL SENDER NAME	E-mail sender name
690	CC_CODE REVIEW CUSTOMER EMAIL SENDER ADDRESS	E-mail sender address
691	CC_CODE REVIEW CUSTOMER EMAIL REPLY ADDRESS	Reply to e-mail address
692	CC_CODE REVIEW CUSTOMER REJECTED TEMPLATE ID	Rejected customer template
693	CC_CODE REVIEW CUSTOMER VALID TEMPLATE ID	Valid customer template
694	CC_CODE_EVENT_AVAILABILITY_HIDE_ADDITIONAL_CAPACITY	Hide additional managed resource capacity
695	CC_CODE_HIDE_TKT_LK_UP_TICKET_GUEST_NAME	Hide Ticket Lookup Guest Name
696	CC_CODE DISPLAY GUEST NAME	Customer display guest name
697	CC_CODE DISPLAY TRANSACTION PROMPT	Customer display transaction prompt
698	CC_CODE CONTACT SYSTEM LOGON EMAIL TEMPLATE ID	Contact Login Email Template
699	CC_CODE AUTO START DBHEARTBEAT	Start DBHeartbeat service at startup
700	CC_CODE_ACTIVITY_TYPE	Lookup Table for Activity types
701	CC_CODE_ACTIVITY_STATE	Lookup Table for Activity States
702	CC_CODE_TRAN_SEAT_SELECT_PNO	Transportation seat selection printer number
703	CC_CODE_TRAN_SEAT_SELECT_TKTSET	Transportation seat selection ticket set
704	CC_CODE_TABLES_CENTRAL_SIAEOPTIONS	SIAE (Italian Tax Agency) Options are managed in central maintenance
705	CC_CODE_LOAD_CUSTOMER_ON_RETURN	Load customer on return
706	CC_CODE_CONTACT_PHONENUMBER_DB_SOURCE	Data Source for Contact Phone Numbers
707	CC_CODE_PASSWORD_ENCRYPTION_METHOD	Password Encryption Method
708	CC_CODE_INTERNAL_ENCRYPTION_BASE_VERSION	Internal encryption base version
709	CC_CODE_HIDE_EVENTS_WITH_INSUFFICIENT_QUANTITY	Hide events with insufficient quantity
710	CC_CODE_ROUND_DENOMINATION_FOP	Rounding - Base Currency - Round FOP
711	CC_CODE_BATCH_SETTLEMENT_URL	Host URL for batch settlement
712	CC_CODE_BATCH_SETTLEMENT_RECORDS_PER_BATCH	Max number of records per settlement batches
713	CC_CODE_BATCH_SETTLEMENT_LOGGING	Advanced logging enabled for batch settlement
714	CC_CODE_BATCH_SETTLEMENT_NODE	Node used for batch settlement
715	CC_CODE_TABLES_CENTRAL_ACCESS_CODE_OVERRIDES	[Not currently in use]

716	CC_CODE_TABLES_CENTRAL_SIAE_VENUES	SIAE (Italian Tax Agency) Venues are managed in central maintenance
717	CC_CODE_TABLES_CENTRAL_SIAE_REDUCTION_CODES	SIAE (Italian Tax Agency) Reduction Codes are managed in central maintenance
718	CC_CODE_TABLES_CENTRAL_SIAE_ITEMS	SIAE (Italian Tax Agency) Items are managed in central maintenance
719	CC_CODE_TABLES_CENTRAL_SIAE_EVENT_ORGANIZERS	SIAE (Italian Tax Agency) Event Organizers are managed in central maintenance
720	CC_CODE_TABLES_CENTRAL_SIAE_EVENTS	SIAE (Italian Tax Agency) Events are managed in central maintenance
721	CC_CODE_PASSWORD_CHANGE_REQUIRED	Users - Security - Password change required on first logon
722	CC_CODE_UPPERCASE_PASS_FIRST_CHAR	Uppercase the first character for pass demographics
723	CC_CODE_GALAXY_SITEID	Galaxy Site ID of the attraction
724	CC_CODE_INCLUDE_DISCOUNT_IN_UPGRADE	Include the discount amount in returns for upgrade
725	CC_CODE_ALLOW_DRAWEROVER_POPUP	
726	CC_CODE_MULTI_SITE_MODULE_ENABLE	Galaxy - internal use only
727	CC_CODE_UPSELL_REPLACEMENT_SCRIPT_TEMPLATE_ID	Web template ID for upsell replacement script
728	CC_CODE_UPSELL_ADD_ON_SCRIPT_TEMPLATE_ID	Web template ID for upsell add-on script
729	CC_CODE_UPSELL_REPLACEMENT_HEADER_SCRIPT_TEMPLATE_ID	Web Template ID for upsell replacement header script
730	CC_CODE_UPSELL_ADD_ON_HEADER_SCRIPT_TEMPLATE_ID	Web Template ID for upsell add-on header script
731	CC_CODE_SHIFT_UPSELL_SUMMARY	Upsell Summary Shift Report
732	CC_CODE_POS_DEFAULT_GIFT_AID_DONATION	Default Gift Aid donation
733	CC_CODE_POS_GIFT_AID_CLAIM_DEFAULT_TAX_RATE	Gift Aid Claim Default Tax Rate
734	CC_CODE_TABLES_CENTRAL_TRANSACTIONAL_UPSELL_OPTIONS	Transactional upsell options are managed in central maintenance
735	CC_CODE_TRANSACTIONAL_UPSELL_REPLACEMENT_SCRIPT_TEMPLATE_ID	Transactional upsell replacement script template ID
736	CC_CODE_TRANSACTIONAL_UPSELL_REPLACEMENT_HEADER_SCRIPT_TEMPLATE_ID	Transactional upsell replacement header template ID
737	CC_CODE_DUPLICATE_CHECK_CUSTCONTACTS	Check for duplicates when saving Contacts
738	CC_CODE_POS_GIFT_AID_PRINTER	Gift Aid statement printer
739	CC_CODE_POS_GIFT_AID_TKTSET	Gift Aid statement ticket set (with donor)
740	CC_CODE_POS_GIFT_AID_TKTSET_NO_DONOR	Gift Aid declaration ticket set (no donor)
741	CC_CODE_TABLES_CENTRAL_SIAE_VOID_REASONS	SIAE Void Reasons
742	CC_CODE_JNL_DELETE	Delete Journal Records
743	CC_CODE_JNL_DELETE_RECORD_AGE	Delete Journal Records Age In Days
744	CC_CODE_TABLES_CENTRAL_SIAE_CALENDARS	
745	CC_CODE_TABLES_CENTRAL_SIAE_CALENDARDETAILS	
746	CC_CODE_DONATION_PLU	
747	CC_CODE_DONATION_PROMPT	
748	CC_CODE_POS_GIFT_AID_COMPANY	POS - Gift Aid - Account Company
749	CC_CODE_POS_GIFT_AID_CATEGORY	POS - Gift Aid - Account Category
750	CC_CODE_POS_GIFT_AID_SUBCAT	POS - Gift Aid - Account SubCategory
751	CC_CODE_USER_LOGON_CARD_PRINTER	Users - Logons - logon card printer
752	CC_CODE_USER_LOGON_CARD_TKTSET	Users - Logons - logon card ticket set
753	CC_CODE_KEEP_ORIG_PASS_UNTIL_EXP_DATE	Pass/Membership option to keep original pass until expiration when renewing
754	CC_CODE_AGENCY_MANUAL_TICKET_REPORT	Agency Reports - Manual Ticket Report
755	CC_CODE_SHIFT_MANUAL_TICKET_REPORT	Shift Reports - Manual Ticket Report
756	CC_CODE_PASS_RENEWAL_EXPIRATION_WINDOW_DAYS	Allow a user to set the window (measured in days and going back in time from the expiration date of a pass) during which a pass may be renewed with a discount.
757	CC_CODE_KEEP_SAME_PASS_ACCOUNT	Keep Same Pass Account - Controls whether or not to keep the same pass account number for the life of the pass (after reissues, renewals, and upgrades)
758	CC_CODE_SIAE_CARD_MANAGER_SERVICE_URL	SIAE Card Manager Service URL - SIAE Card Manager Service URL used by Galaxy to send requests to get fiscal seals
759	CC_CODE_PASS_PORTAL_PAYMENT_CONTRACT_ITEM_GROUP	Pass Portal Payment Contract Item - In the Make Payment operation of the Pass Portal, this Item Group defines the PLUs that the user may choose from.
760	CC_CODE_SHOW_ONLY_CURRENT_PASSES	Show only current passes when browsing
761	CC_CODE_PASS_PORTAL_REISSUE_ITEM_GROUP	Pass Portal reissue item group
762	CC_CODE_ENABLE_DYNAMIC_CURRENCY_CONVERSION	Enable Dynamic Currency Conversion
763	CC_CODE_DCC_URL	Dynamic Currency Conversion URL
764	CC_CODE_DCC_MERCHANT_ID	Dynamic Currency Conversion Merchant ID
765	CC_CODE_DCC_ACQUIRER	Dynamic Currency Conversion Acquirer
766	CC_CODE_TABLES_CENTRAL_CURRENCIES	Use Central Dynamic Currency Conversion Table
767	CC_CODE_ENABLE_REFUND_REASON_BY_FOP	Enable refund reasons by FOP
768	CC_CODE_NOT_USED_2	Not used
769	CC_CODE_USER_SAVE_LOGGING	User Save Logging
770	CC_CODE_PTO_PREVENT_MP_MODE	Force MP Mode
771	CC_CODE_AUTO_CAPITALIZE_PLU	Auto Capitalize PLU
772	CC_CODE_DISABLE_QOP_ORDER_SEARCH	Disable quick order pickup search by order
773	CC_CODE_DISABLE_QOP_CC_SEARCH	Disable quick order pickup search by credit card
774	CC_CODE_DISABLE_QOP_CONTACT_SEARCH	Disable quick order pickup search by contact
775	CC_CODE_TICKET_ACTIVATE_PNO	
776	CC_CODE_TICKET_ACTIVATE_TKTSET	
777	CC_CODE_TICKET_ACTIVATION_STATEMENT_ID	
778	CC_CODE_HIDE_TKT_LK_UP_CREDIT_CARD	Hide Ticket Lookup Credit Card Information
779	CC_CODE_HIDE_TKT_LK_UP_CREDIT_CARD_ORDER_ID	Hide Ticket Lookup Credit Card Order ID
780	CC_CODE_HIDE_TKT_LK_UP_CREDIT_CARD_NUMBER	Hide Ticket Lookup Credit Card Number

781	CC_CODE_HIDE_TKT_LK_UP_CREDIT_CARD_CARD HOLDER_NAME	Hide Ticket Lookup Credit Card Holder Name
782	CC_CODE_HIDE_TKT_LK_UP_CREDIT_CARD_CARD_TYPE	Hide Ticket Lookup Credit Card Card Type
783	CC_CODE_REQUIRE_CONTACT_LOOKUP_FOR_NEW_MEMBER	Require contact lookup for new members
784	CC_CODE_USE_WINDOWS_REPORT_PRINTER	Use Windows report printer
785	CC_CODE_WINDOWS_REPORT_PRINTER_NAME	Windows report printer name
786	CC_CODE_PASS_PORTAL_PAYMENT_PROCESSOR	Pass Portal Payment Processor
787	CC_CODE_PASS_PORTAL_DISPLAY_EXCHANGE_CARD	Display exchange card in pass portal
788	CC_CODE_PASS_PORTAL_DISPLAY_MERGE	Display merge button in pass portal
789	CC_CODE_PASS_PORTAL_DISPLAY_SPLIT	Display split button in pass portal
790	CC_CODE_PASS_PORTAL_DISPLAY_ADDON	Display add-on button in pass portal
791	CC_CODE_PASS_PORTAL_HIDE_RENEWAL_ANOTHER	Hide renew another button in pass portal
792	CC_CODE_PASS_PORTAL_REISSUE_CARD_ITEM_GROUP	Pass portal reissue member card item group
793	NOT USED	NOT USED
794	CC_CODE_PASS_PORTAL_DISPLAY_NEW_MEMBER	
795	CC_CODE_ENABLE_WEB_STORE_TRANSLATION_SERVICE	Enable web store translation service
796	CC_CODE_TABLES_CENTRAL_DISCOUNT_PRIVILEGE	Use central discount privileges
797	CC_CODE_AUTO_SAVE_SALES_CHANNEL_MANAGER_CHANGES	Automatically save Sales Channel Manager changes when changing view
798	CC_CODE_PROMPT_BEFORE_SAVING_SALES_CHANNEL_MANAGER_CHANGES	Prompt before automatically saving Sales Channel Manager changes
799	CC_CODE_PASS_PORTAL_LOAD_HOUSEHOLD	Load household in pass portal
800	CC_CODE_FORCE_SL_CRITERIA	Enforce system logon criteria
801	CC_CODE_GIFT_CREATION_METHOD	Gift creation method. Determines if there is a single gift per transaction or if there is a gift per item.
802	CC_CODE_AUTO_ASSIGN_PURCHASER	Automatically assign the transaction contact as the pass purchaser when selling a new pass
803	CC_CODE_MEMBERSHIP_SPLIT_RELATIONSHIP_TYPE	Membership split relationship type. Determines replacement relationship type for relationships broken by a membership split.
804	CC_CODE_AUTO_CREATE_RECIPROCAL_RELATIONSHIP	When creating or editing a joint member, automatically create a reciprocal relationship (from primary to the member) of the same relationship type.
805	CC_CODE_PASS_PORTAL_ALLOW_DOWNGRADES	Allow pass downgrades in pass portal
806	CC_CODE_DISPLAY_RE_GIFT_TYPE	Display gift type from The Raiser's Edge in the gift viewer
807	CC_CODE_COMBINE_INVOICE_LINES	Combine invoice statement lines
808	CC_CODE_COMBINE_CREDITMEMO_LINES	Combine credit memo lines
809	CC_CODE_TABLES_CENTRAL_ATTRACTIONS	Use central attractions
810	CC_CODE_GATEWAY_SITE_ID	Gateway site ID
811	CC_CODE_AUTO_SAVE_SALES_CHANNEL_MANAGER_CHANGES	Automatically save Sales Channel Manager changes when changing view
812	CC_CODE_PROMPT_BEFORE_SAVING_SALES_CHANNEL_MANAGER_CHANGES	Prompt before automatically saving Sales Channel Manager changes
813	CC_CODE_PREVENT_NEGATIVE RETURNS	Prevent returns using the minus key
814	CC_CODE_PURCHASER_PICKLIST	Use pass picklist to select purchaser
815	CC_CODE_KIOSK_EXTERNAL_AUTHORIZATION_FOP	FOP to use for credit card authorization on an external device in the Kiosk
816	CC_CODE_TRANSACTION_RECEIPT_PNO	POS - Ticket Sets - Transaction Receipt printer number
817	CC_CODE_TRANSACTION_RECEIPT_TKTSET	POS - Ticket Sets - Transaction Receipt ticket set
818	CC_CODE_ITEM_ATTRIBUTE_GROUP_ID	
819	CC_CODE_COA_ATTRIBUTE_GROUP_ID	
820	CC_CODE_FOP_ATTRIBUTE_GROUP_ID	
821	CC_CODE_CUSTOMER_ATTRIBUTE_GROUP_ID	
822	CC_CODE_NODE_ATTRIBUTE_GROUP_ID	
823	CC_CODE_ACCESS_CODE_ATTRIBUTE_GROUP_ID	
824	CC_CODE_DISCOUNT_ATTRIBUTE_GROUP_ID	
825	CC_CODE_PREVENT_CONCURRENT_LOGONS	Indicates if logons are prevented if the user is logged on somewhere else in the system.
826	CC_CODE_ITEM_FIELD_ATTRIBUTE_GROUP_ID	
827	CC_CODE_EVENT_MANAGEMENT_SERVER_BASE_URL	
828	CC_CODE_DISPLAY_VEHICLE_DETECTOR_STATUS	Display vehicle detector status in POS or ACS.
829	CC_CODE_ENFORCE_ONE TRANSACTION_PER_VEHICLE	Allow only one POS transaction per vehicle.
830	CC_CODE_CREATE_SYSLOG_TRIGGERES	
831	CC_CODE_RECORD_VEHICLE_DETECTOR_USAGE	
832	CC_CODE_VEHICLE_DETECTOR_ENTRY_ACCESS_CODE	
833	CC_CODE_VEHICLE_DETECTOR_EXIT_ACCESS_CODE	
834	CC_CODE_SESELLER_CHARGE_ACCOUNT_FOP	
835	CC_CODE_USE_AVIS_ZIP_FOR_TRANSACTION_PROMPT	Use the AVS zip code instead of using the transaction prompt to collect the zip code for AVS credit card transactions.
836	CC_CODE_DUAL_RELATIONSHIP_TYPE_ID	Dual relationship type ID
837	CC_CODE_FORMER_DUAL_RELATIONSHIP_TYPE_ID	Former dual relationship type ID
838	CC_CODE_ENABLE_SEAT_OPTIONS	
839	CC_CODE_NUMBER_OF_SEAT_OPTIONS	
840	CC_CODE_MIN_PERCENT_FOR_SEAT_OPTIONS	
841	CC_CODE_DEFAULT_MAX_RESERVED_SEATS_PER_SESSION	
842	CC_CODE_RS_ITEM_GROUP_ATTRIBUTE_ID	
843	CC_CODE_ENABLE_SINGLE_SEAT_CHECK	
844	CC_CODE_DEFAULT_OE_CONTACT_TYPE_FOR_CONTACT_REQ_PLUS	
845	CC_CODE_LICENSE_SERVER_ENDPOINT	
846	CC_CODE_HIDE_PASS_FIELDS	
847	CC_CODE_SHOW_ONLY_HISTORY_FOR_CURRENT_PASS	Show only the pass history for the current pass record. Do not show history for prior passes in the

		chain
848	CC_CODE_UPSELL_SELECT_QUANTITY	Allow selection of quantity for upsell options in POS and Order Entry.
849	CC_CODE_ALLOWMAILINGS_DEFAULT_UNCHECKED	
850	CC_CODE_ALLOWEMAIL_DEFAULT_UNCHECKED	
851	CC_CODE_USE_UNICODE_STRINGS,	
852	CC_CODE_STRATUS_XML_AUTHORIZATION_ONLY,	
853	CC_CODE_DEFAULT_ITEM_FIELD_ATTRIBUTE_GROUP_ID	
854	CC_CODE_POS_CUST_TRANS_IGNORE_ITEM_FOP	Ignore the item FOP configuration when determining available FOPs in a POS customer transaction.
855	CC_CODE_ALLOW_SKIP_POS_GUEST_PHOTO_PROMPT	Allow POS guest photo prompt to be skipped
856	CC_CODE_AUTO_POPULATE_ROSTERS	Auto populate additional rosters
857	CC_CODE_POS_PROMPT_DISPLAY_POLICY	Designates policy for how item selection prompts are displayed in POS
858	CC_CODE_NOT_USED_17	Not used
859	CC_CODE_IGNORE_DISCOUNT_FOR_PASS_UPGRADE	Ignores discount amount from original pass sale for pass upgrade returns.
860	CC_CODE_GUEST_NAME_PROMPT_POLICY	
861	CC_CODE_JOURNAL_PURGE_TEST_MODE	
862	CC_CODE_TABLES_CENTRAL_MODIFIERS	
863	CC_CODE_SET_GALAXY_COLOR_SCHEME	
864	CC_CODE_ONLINE_EXCHANGE_ALLOWANCE	
865	CC_CODE_ONLINE_EXCHANGE_ALLOWED_DAYS_BEFORE_VISIT_DATE	
866	CC_CODE_TABLES_CENTRAL_MODIFIER_GROUPS	
867	CC_CODE_TABLES_CENTRAL_ITEM_MODIFIER_GROUP_DETAILS	
868	CC_CODE_EGALAXY_ENFORCE_SOURCE_ID	eGalaxy - Enforce SourceID
869	CC_CODE_EGALAXY_SIAE_SMART_CARD_PIN	eGalaxy - SIAE Smart Card PIN
870	CC_CODE_EGALAXY_SESSION_USE_UNIVERSAL_TIME	eGalaxy - Session Use Universal Time
871	CC_CODE_EGALAXY_SSL_ENABLE_SSL	eGalaxy - Enable SSL
872	CC_CODE_EGALAXY_SSL_ROOT_CERT_FILE	eGalaxy - SSL Root Cert File
873	CC_CODE_EGALAXY_SSL_CERT_FILE	eGalaxy - SSL Cert File
874	CC_CODE_EGALAXY_SSL_KEY_FILE	eGalaxy - SSL Key File
875	CC_CODE_EGALAXY_SSL_PASSWORD	eGalaxy - SSL Key File Password
876	CC_CODE_EGALAXY_SSL_PORT	eGalaxy - SSL Port
877	CC_CODE_EGALAXY_PAYMENT_PROCESSOR_URL	eGalaxy - Payment Processor URL
878	CC_CODE_EGALAXY_AUDIT_ADD_CUSTOMERS	eGalaxy - Audit Add Customers
879	CC_CODE_EGALAXY_AUDIT_EDIT_CUSTOMERS	eGalaxy - Audit Edit Customers
880	CC_CODE_EGALAXY_AUDIT_DELETE_CUSTOMERS	eGalaxy - Audit Delete Customers
881	CC_CODE_EGALAXY_AUDIT_ADD_ORDERS	eGalaxy - Audit Add Orders
882	CC_CODE_EGALAXY_AUDIT_EDIT_ORDERS	eGalaxy - Audit Edit Orders
883	CC_CODE_EGALAXY_AUDIT_DELETE_ORDERS	eGalaxy - Audit Delete Orders
884	CC_CODE_EGALAXY_AUDIT_ADD_ORDER_LINES	eGalaxy - Audit Add Order Lines
885	CC_CODE_EGALAXY_AUDIT_EDIT_ORDER_LINES	eGalaxy - Audit Edit Order Lines
886	CC_CODE_EGALAXY_AUDIT_DELETE_ORDER_LINES	eGalaxy - Audit Delete Order Lines
887	CC_CODE_EGALAXY_AUDIT_ADD_ACCESS_CODES	eGalaxy - Audit Add Access Codes
888	CC_CODE_EGALAXY_AUDIT_EDIT_ACCESS_CODES	eGalaxy - Audit Edit Access Codes
889	CC_CODE_EGALAXY_AUDIT_DELETE_ACCESS_CODES	eGalaxy - Audit Delete Access Codes
890	CC_CODE_EGALAXY_AUDIT_ADD_EVENTS	eGalaxy - Audit Add Events
891	CC_CODE_EGALAXY_AUDIT_EDIT_EVENTS	eGalaxy - Audit Edit Events
892	CC_CODE_EGALAXY_AUDIT_DELETE_EVENTS	eGalaxy - Audit Delete Events
893	CC_CODE_EGALAXY_AUDIT_ADD_ITEMS	eGalaxy - Audit Add Items
894	CC_CODE_EGALAXY_AUDIT_EDIT_ITEMS	eGalaxy - Audit Edit Items
895	CC_CODE_EGALAXY_AUDIT_DELETE_ITEMS	eGalaxy - Audit Delete Items
896	CC_CODE_EGALAXY_AUDIT_ADD_LOCKOUTS	eGalaxy - Audit Add Lockouts
897	CC_CODE_EGALAXY_AUDIT_EDIT_LOCKOUTS	eGalaxy - Audit Edit Lockouts
898	CC_CODE_EGALAXY_AUDIT_DELETE_LOCKOUTS	eGalaxy - Audit Delete Lockouts
899	CC_CODE_EGALAXY_AUDIT_ADD_PASSES	eGalaxy - Audit Add Passes
900	CC_CODE_EGALAXY_AUDIT_EDIT_PASSES	eGalaxy - Audit Edit Passes
901	CC_CODE_EGALAXY_AUDIT_DELETE_PASSES	eGalaxy - Audit Delete Passes
902	CC_CODE_EGALAXY_AUDIT_ADD_WAIT_LISTS	eGalaxy - Audit Add Wait Lists
903	CC_CODE_EGALAXY_AUDIT_EDIT_WAIT_LISTS	eGalaxy - Audit Edit Wait Lists
904	CC_CODE_EGALAXY_AUDIT_DELETE_WAIT_LISTS	eGalaxy - Audit Delete Wait Lists
905	CC_CODE_WEB_ORDER_PROCESSOR_INTERVAL	Web Order Processor - Interval
906	CC_CODE_WEB_ORDER_PROCESSOR_SCAN_COUNT	Web Order Processor - Scan Count
907	CC_CODE_WEB_ORDER_PROCESSOR_PRINT_AT_HOME_FOLDER	Web Order Processor - Print at Home Folder
908	CC_CODE_WEB_ORDER_PROCESSOR_DELETE_TEMP_PDF	Web Order Processor - Delete Temp PDF
909	CC_CODE_WEB_ORDER_PROCESSOR_LOG_ORDER_PROCESSING_DETAILS	Web Order Processor - Log Order Processing Details
910	CC_CODE_ORDER_PICKUP_CUSTOM_USER_AGENT	Order Pickup - Custom User Agent
911	CC_CODE_ORDER_PICKUP_USE_PROXY_SERVER	Order Pickup - Use Proxy Server
912	CC_CODE_PASS_NOTES_MIXED_MODE	
913	CC_CODE_DISABLE_QOP_CONTACT_IDENTITY_NUMBER_SEARCH	
914	CC_CODE_EGALAXY_EVENT_MANAGEMENT_SERVER_USERNAME	eGalaxy - Event Management Server Username
915	CC_CODE_EGALAXY_EVENT_MANAGEMENT_SERVER_PASSWORD	eGalaxy - Event Management Server Password

916	CC_CODE_EGALAXY_EVENT_MANAGEMENT_SERVER_ENCRYPTION_VERSION	eGalaxy - Event Management Server Encryption Version
917	CC_CODE_DISABLE_OE_EXTERNAL_ORDER_ID_EDIT	Disable External Order ID Editing in Order Entry
918	CC_CODE_TRANSACTION_LOOKUP_BARCODE_MEDIA_DEFINITION	Media definition used to create transaction lookup barcodes
919	CC_CODE_DEPOSIT_ID_MIN_LENGTH	Deposit ID Min Length
920	CC_CODE_DEPOSIT_ID_MAX_LENGTH	Deposit ID Max Length
921	CC_CODE_MEDIA_DEF_FOR_SERIALIZED_STATEMENTS	Media Definition to be used for serialized statements
922	CC_CODE_DUPLICATE_TEXT_FOR_SERIALIZED_STATEMENTS	String defining how the word "DUPLICATE" will be rendered
923	CC_CODE_SHOW_TKT_LK_UP_ONLINE_TICKET_EXCHANGE	Determine whether to show online ticket exchange tab in ticket lookup window
924	CC_CODE_MEDIA_DEF_FOR_SERIALIZED_CREDITMEMOS	Media Definition to be used for serialized credit memo statements
925	CC_CODE_ENCRYPT_SENSITIVE_CONTACT_FIELDS	Encrypt sensitive contact fields
926	CC_CODE_APPEAL_CODE_REQUIRED	Appeal code required
927	CC_CODE_SERVICES_CIPHER_LIST	Cipher list used for secure communications by services
928	CC_CODE_PRIVATE_STATE_COLOR	Background cell color for the Private state column in the Bateaux event screen
929	CC_CODE_PRIVATE_STATE_FONT_COLOR	Font color for the Private state column in the Bateaux event screen
930	CC_CODE_EVENT_RESERVATION_COMPLETE_COLOR	Background row color for completed Event Reservations
931	CC_CODE_EVENT_RESERVATION_COMPLETE_SELECTED_COLOR	Background row color for completed and selected Event Reservations
932	CC_CODE_EVENT_RESERVATION_INCOMPLETE_COLOR	Background row color for incomplete Event Reservations
933	CC_CODE_EVENT_RESERVATION_INCOMPLETE_SELECTED_COLOR	Background row color for incomplete and selected Event Reservations
934	CC_CODE_EVENT_RESERVATION_FONT_COLOR	Font color for the Event Reservations screen
935	CC_CODE_DEACTIVATED_STATE_COLOR	Deactivated color in state column for Bateaux
936	CC_CODE_DEACTIVATED_STATE_FONT_COLOR	Deactivated font color in state column for Bateaux
937	CC_CODE_OFF_SALE_STATE_COLOR	Off-Sale color in state column for Bateaux
938	CC_CODE_OFF_SALE_STATE_FONT_COLOR	Off-Sale font color in state column for Bateaux
939	CC_CODE_COLLECTENDORSEMENT_BINS	True if Galaxy should collect the BIN from endorsements, False if not.
940	CC_CODE_TABLES_CENTRAL_PRICE_PROGRAM_GROUPS	Use central price program groups
941	CC_CODE_DUAL_MEMBERSHIP	Dual Membership Mode
942	CC_CODE_BIN_LOOKUP_BINDB_URL	URL for BINDB API service
943	CC_CODE_BIN_LOOKUP_API_KEY	BINDB API key
944	CC_CODE_BIN_LOOKUP_POLL_INTERVAL	BIN Lookup Service poll interval. The amount of time (seconds) between checks for new transactions with BINs that need to be looked up in BINDB.
945	CC_CODE_BIN_LOOKUP_SCAN_COUNT	BIN Lookup Service scan count. The maximum number of records that will be processed each time the service checks for new transactions.
946	CC_CODE_BIN_LOOKUP_USE_PROXY_SERVER	BIN Lookup Service use proxy server. Whether or not to use a proxy server.
947	CC_CODE_BIN_LOOKUP_BIN_STALE_INTERVAL	Time (hours) within which we will not look up the same BIN twice
948	CC_CODE_WEB_CART_TIME_ZONE_OFFSET_HR	Web Cart Time Zone Offset (Hours)
949	CC_CODE_LICENSE_SERVER_KEY	Customer License Key (GUID) for Galaxy Management Server Features
950	CC_CODE_EVENT_RESERVATION_TABLE_ATTRIBUTE_ID	Attribute that represents a Table on Event Reservation orders
951	CC_CODE_ENFORCE_CREDIT_LIMIT_BY_CONTACT	Enforce credit limit by contact instead of by customer
952	CC_CODE_INCLUDE_COMPANY_IN_SERIALIZED_STATEMENTS	Include company in Serialized Statement serial numbers
953	CC_CODE_KIOSK_MAX_ORDER_REPRINTS	Maximum number of times an order can be reprinted at the kiosk (0 = unlimited)
954	CC_CODE_KIOSK_PREVENT_EXPIRED_TICKET_ORDER_REPRINT	Prevent orders with expired tickets from being reprinted at the kiosk
955	CC_CODE_KIOSK_PREVENT_USED_TICKET_ORDER_REPRINT	Prevent orders with used tickets from being reprinted at the kiosk
956	CC_CODE_KIOSK_ORDER_REPRINT_USED_TICKET_EXCLUDE_ITEM_GROUP_ID	Item group to exclude when preventing orders with expired tickets from being reprinted at the kiosk
957	CC_CODE_CREATE_NEW_ORDER_FOR_SECURE_ORDER RETURNS	Create new order for secure order returns
958	CC_CODE_ENABLE_GUEST_SALE_LIMITS	Enable Guest Sale Limits
959	CC_CODE_VOID_PASS_WITHOUT_VALIDATION	Allow voiding of passes without validation (hidden option)
960	CC_CODE_DONT_USE_TLS_10	Prevent the use of TLS v1.0
961	CC_CODE_DONT_USE_TLS_11	Prevent the use of TLS v1.1
962	CC_CODE_INITIAL_GIFT_AID_PROMPT_TEMPLATE_ID	Initial Gift Aid prompt template ID
963	CC_CODE_GIFT_AID_DONOR_PROMPT_TEMPLATE_ID	Gift Aid donor prompt template ID
964	CC_CODE_NON_GIFT_AID_COMPANY	Non Gift Aid company
965	CC_CODE_NON_GIFT_AID_CATEGORY	Non Gift Aid category
966	CC_CODE_NON_GIFT_AID_SUBCAT	Non Gift Aid subcategory
967	CC_CODE_HIDE_NON_GIFT_AID	Hide Non Gift Aid button on prompt
968	CC_CODE_GIFT_AID_CONTACT_FIELD_ATTRIBUTE_GROUP_ID	Gift Aid contact field attribute group ID
969	CC_CODE_DISABLE_GIFT_AID_COLLECT_LATER	Disable Gift Aid Collect Later Option
970	CC_CODE_GIFT_AID_SURVEY_MINIMUM	Minimum Gift Aid items to display survey
971	CC_CODE_DEFAULT_PASS_LOOKUP_BY_ACCOUNT	Default to lookup passes by account (Pass Portal, pass-required)
972	CC_CODE_GIFT_AID_SURVEY_ID	Gift Aid survey ID
973	CC_CODE_GIFT_AID_EMAIL_SENDER_NAME	Gift Aid Email sender name
974	CC_CODE_GIFT_AID_EMAIL_SENDER_ADDRESS	Gift Aid Email sender address
975	CC_CODE_GIFT_AID_EMAIL_REPLY_TO_ADDRESS	Gift Aid Email reply to address
976	CC_CODE_GIFT_AID_EMAIL_OPTION_TEMPLATE_ID	Gift Aid Email prompt template ID
977	CC_CODE_GIFT_AID_EMAIL_TEMPLATE_ID	Gift Aid Email Template ID
978	CC_CODE_GIFT_AID_ROUND_AMOUNT	Gift Aid round amount
979	CC_CODE_FISCAL_YEAR_END_MONTH	Fiscal year end month
980	CC_CODE_FISCAL_YEAR_END_DAY	Fiscal year end day
981	CC_CODE_GIFT_AID_DONOR_TYPE_PROMPT_TEMPLATE_ID	Gift Aid donor type prompt template ID
982	CC_CODE_GIFT_AID_DONOR_TYPE_ID	Gift Aid donor type ID

983	CC_CODE_GIFT_AID_DEFAULT_TO_ORDER_CONTACT	Default the Gift Aid contact to the order contact
984	CC_CODE_PICK_PASS_SHOW_FIRST_NAME	Show First Name field in pass search results
985	CC_CODE_PICK_PASS_SHOW_MIDDLE_NAME	Show Middle Name field in pass search results
986	CC_CODE_PICK_PASS_SHOW_LAST_NAME	Show Last Name field in pass search results
987	CC_CODE_PICK_PASS_SHOW_PASS_ACCOUNT	Show Pass Account field in pass search results
988	CC_CODE_PICK_PASS_SHOW_PASS_KIND	Show Pass Kind field in pass search results
989	CC_CODE_PICK_PASS_SHOW_DOB	Show DOB field in pass search results
990	CC_CODE_PICK_PASS_SHOW_STREET1	Show Street Address Line 1 field in pass search results
991	CC_CODE_PICK_PASS_SHOW_STREET2	Show Street Address Line 2 field in pass search results
992	CC_CODE_PICK_PASS_SHOW_STREET3	Show Street Address Line 3 field in pass search results
993	CC_CODE_PICK_PASS_SHOW_CITY	Show City field in pass search results
994	CC_CODE_PICK_PASS_SHOW_STATE	Show State field in pass search results
995	CC_CODE_PICK_PASS_SHOW_ZIP	Show Zip Code field in pass search results
996	CC_CODE_PICK_PASS_SHOW_PHONE	Show Phone Number field in pass search results
997	CC_CODE_PICK_PASS_SHOW_EMAIL	Show Email Address field in pass search results
998	CC_CODE_PICK_PASS_SHOW_GENDER	Show Gender field in pass search results
999	CC_CODE_PICK_PASS_SHOW_EXPIRATION_DATE	Show Pass Expiration Date field in pass search results
1000	CC_CODE_BLANK_PASS_VISUAL_ID_LOGGING	Log changes that blank the pass visual ID
1001	CC_CODE_INCREMENT_SERIAL_FOR_EACH_GENERATED_STATEMENT	Increment the serial number for each generated serialized statement
1002	CC_CODE_GALAXY_CONNECT_URL	The Galaxy Connect URL
1003	CC_CODE_CONNECTOR_URL	The Connect proxy URL
1004	CC_CODE_GALAXY_CONNECT_ITEM_GROUP	Items to process through Galaxy Connect
1005	CC_CODE_KIOSK_DISABLE_SWIPE_ON_IDLE_SCREEN	Kiosk disable credit card swipe at Idle screen
1006	CC_CODE_KEEP_ORIGINAL_ACCOUNT_ON_UPGRADE	Keep original account on upgrade
1007	CC_CODE_PICK_PASS_SHOW_UDF_01	Show the UDF01 column
1008	CC_CODE_PICK_PASS_SHOW_UDF_02	Show the UDF02 column
1009	CC_CODE_PICK_PASS_SHOW_UDF_03	Show the UDF03 column
1010	CC_CODE_PICK_PASS_SHOW_UDF_04	Show the UDF04 column
1011	CC_CODE_PICK_PASS_SHOW_UDF_05	Show the UDF05 column
1012	CC_CODE_PICK_PASS_SHOW_UDF_06	Show the UDF06 column
1013	CC_CODE_PICK_PASS_SHOW_UDF_07	Show the UDF07 column
1014	CC_CODE_PICK_PASS_SHOW_UDF_08	Show the UDF08 column
1015	CC_CODE_PICK_PASS_SHOW_UDF_09	Show the UDF09 column
1016	CC_CODE_PICK_PASS_SHOW_UDF_10	Show the UDF10 column
1017	CC_CODE_PICK_PASS_SHOW_BARCODE	Show the Barcode column
1018	CC_CODE_PICK_PASS_SHOW_ID_NUMBER	Show the ID Number column
1019	CC_CODE_PICK_PASS_SHOW_COUNTRY_CODE	Show the Country Code column
1020	CC_CODE_PICK_PASS_SHOW_OPEN_DATE	Show the Open Date column
1021	CC_CODE_PICK_PASS_SHOW_LAST_UPDATE_DATE	Show the Last Update Date column
1022	CC_CODE_PICK_PASS_SHOW_STATUS	Show the Status column
1023	CC_CODE_PICK_PASS_SEARCH_UDF_NAME	Custom name for the UDF search criterion
1024	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_01	Custom name for the UDF01 column
1025	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_02	Custom name for the UDF02 column
1026	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_03	Custom name for the UDF03 column
1027	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_04	Custom name for the UDF04 column
1028	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_05	Custom name for the UDF05 column
1029	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_06	Custom name for the UDF06 column
1030	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_07	Custom name for the UDF07 column
1031	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_08	Custom name for the UDF08 column
1032	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_09	Custom name for the UDF09 column
1033	CC_CODE_PICK_PASS_SEARCH_UDF_FIELD_NAME_10	Custom name for the UDF10 column
1034	CC_CODE_GENERATE_DETAILS_FOR_SERIALIZED_STATEMENTS	True if Galaxy should generate SerializedStatementDetails records when generating a Serialized Statement.
1035	CC_CODE_POS_DEFAULT_TO_FIRST_CUST_SP	Use the first customer sales program in POS if no default is specified.
1036	CC_CODE_SPECIAL_NEEDS_KEYWORD_VALUE_IF_CONTACT_FLAG_SET	Value used for @CONTACT_SPECIAL_NEEDS@PASS_CONTACT_SPECIAL_NEEDS keywords if contact's "special needs" flag is set
1037	CC_CODE_SPECIAL_NEEDS_KEYWORD_VALUE_IF_CONTACT_FLAG_UNSET	Value used for @CONTACT_SPECIAL_NEEDS@PASS_CONTACT_SPECIAL_NEEDS keywords if contact's "special needs" flag is not set
1038	CC_CODE_SPECIAL_NEEDS_CAPTION	Caption value for the 'Special Needs' check box on system forms
1039	CC_CODE_CONTACT_COUNTRY_CODE_DEFAULT	Default value to be used for the country code for all new contacts.
1040	CC_CODE_DISPLAY_SEND_PASS_TO	Indicates that the 'Send To' option will be shown on the pass screen.
1041	CC_CODE_NOT_USED_21	Not used.
1042	CC_CODE_SERIALIZED_STATEMENT_COMPANY_SELECTION_MODE	Indicates how the company will be determined for Serialized Statements
1043	CC_CODE_ENABLE_SERIALIZED_STATEMENTS_FEATURE	Indicates if the Serialized Statements feature is enabled
1044	CC_CODE_EGALAXYCONFIG_ERROREMAILTEMPLATEID	WebTemplate ID for the eGalaxy Server Error Email template
1045	CC_CODE_TABLES_CENTRAL_ATTRIBUTES	Use central attributes
1046	CC_CODE_TABLES_CENTRAL_GENERIC_CALENDARS	Use central calendars
1047	CC_CODE_EXPIRE_MEMBER_BENEFITS_ON_RENEWAL	Sets the expiration date on old member benefits when renewing a pass to the same pass record
1048	CC_CODE_HONOR_EOD_FOR_JOURNAL_ACCOUNT_DATE	Honor End Of Day time when journalizing Account Date

1049	CC_CODE_JOURNAL_SENDER_SERVICE_RESTART_MAX_DELAY	Maximum number of seconds to wait before restarting journal-based sender services after database reconnection
1050	CC_CODE_KIOSK_FOP_GROUP	FOP group used to determine available FOPs for the HTML kiosk
1051	CC_CODE_SMART_UPSELL_ENABLED	Determines if the "smart upsell" feature is enabled
1052	CC_CODE_MASK_VISUAL_ID	Mask visual IDs
1053	CC_CODE_REQUIRE_APPROVAL_FOR_FAILED_CASHOUT_BALANCE	Require supervisor approval after failed cashout due to incorrect balance.
1054	CC_CODE_PREVENT_REPRINT_OF_NON_ORDER_TICKETS	Prevent reprint of tickets from outside the original order.
1056	CC_CODE_DISABLE_AUTOMATIC_SECURE_FLAG_CHANGES	Disable the automatic changing of the "Secure" checkbox in Order Entry when setting the status of an order.
1057	CC_CODE_MASK_OF_EXTERNAL_ID	Mask Order Entry external IDs.
1058	CC_CODE_POPULATE_EVENTS_IN_PACKAGE	Cascade selected event to package details with the same EventType and Resource.
1059	CC_CODE_MASK_OF_EXTERNAL_ID_FIRST_LENGTH	Mask OE External ID - show first x characters length.
1060	CC_CODE_MASK_OF_EXTERNAL_ID_LAST_LENGTH	Mask OE External ID - show last y characters length
1061	CC_CODE_DISABLE_QOP_ORDER_SEARCH_BY_ORDERID	Disable quick order pickup order search by order ID
1062	CC_CODE_DISABLE_QOP_ORDER_SEARCH_BY_EXTERNALID	Disable quick order pickup order search by external ID
1063	CC_CODE_DISABLE_QOP_ORDER_SEARCH_BY_REFERENCE	Disable quick order pickup order search by reference
1064	CC_CODE_LOG_PRINTING_FISCAL_REPORTS	Log printing of fiscal reports
1068	CC_CODE_DISPLAY_CONFIRM_ORDER_PROMPT	Display the Confirm Order prompt during Quick Order Pickup
1069	CC_CODE_DISPLAY_IDENTIFICATION_PROMPT	Display the Confirm Identification prompt during Quick Order Pickup
1070	CC_CODE_ALLOW_QOP_PARTIAL_SEARCH_BY_EXTERNAL_ID	Allow partial searches by External ID during Quick Order Pickup
1071	CC_CODE_SKIP_AUTO_ISSUANCE_AT_QOV	When in Quick Order Validation, determines whether we automatically issue when a Single order is found, or display the Multiple Order selection screen
1072	CC_CODE_DEFAULT_CUSTOMER_EXTERNAL_ACCOUNT_ID	Customer external account ID to use when one isn't configured on customer.
1073	CC_CODE_ALLOW_AMOUNT_RESTRICTIONS_FOR_MP RETURNS	Allow amount restrictions when performing multi-payment returns
1074	CC_CODE_ALLOW_INACTIVE_UPGRADES	Allow upgrades of inactive media
1075	CC_CODE_ALLOW_INACTIVE RETURNS	Allow returns of inactive media
1076	CC_CODE_ALLOW_INACTIVE_ADD_OONS	Allow entitlement add ons to inactive media
1077	CC_CODE_TABLES_CENTRAL_TRANSLATION_FIELDS	Centrally Manage Translation Fields
1078	CC_CODE_TABLES_CENTRAL_TRANSLATION_LANGUAGES	Centrally Manage Translation Languages
1079	CC_CODE_TABLES_CENTRAL_TRANSLATION_TABLES	Centrally Manage Translation Tables
1080	CC_CODE_TABLES_CENTRAL_TRANSLATION_VALUES	Centrally Manage Translation Values
1081	CC_CODE_TABLES_CENTRAL_ITEM_TRANSLATIONS	Centrally Manage Item Translations
1082	CC_CODE_ENABLE_JOURNAL_DIGITAL_SIGNATURES	Enable the generation of digital signatures for journal transactions.
1083	CC_CODE_JOURNALIZE_FISCAL_SUMMARIES	Journalize Fiscal Summaries
1084	CC_CODE_FRENCH_FISCAL_MODE	French Fiscal Mode
1085	CC_CODE_NOT_USED_37	Not used
1086	CC_CODE LET JOURNAL_EXPORTERS_FINISH BEFORE SHUTDOWN	Allow ticket sender and journal sender services to process all pending work before Galaxy shuts down
1087	CC_CODE_PASS_KIND_GROUP_ATTRIBUTE_GROUP_ID	
1088	CC_CODE_DEFAULT_DELIVERY_METHOD	Default delivery method to use in Order Entry
1089	CC_CODE_CUSTOM_FOP_MENU_ID	Custom FOP Menu ID
1090	CC_CODE_TABLES_CENTRAL_FOP_MENU_HEADERS	Centrally Manage FOP Menu Headers Table
1091	CC_CODE_TABLES_CENTRAL_FOP_MENU_DETAILS	Centrally Manage FOP Menu Details Table
1092	CC_CODE_ACS_RESERVATION_CONFIRMATION_SCRIPT_TEMPLATE_ID	ACS Reservation Confirmation Script Template ID
1093	CC_CODE_ACS_RESERVATION_SERVER_BASE_URL	ACS Reservation Server URL
1094	CC_CODE_ALLOW_LEGACY_QOV_PROCESSING	Allow Legacy QOV Processing
1095	CC_CODE_TABLES_INSERT_ACS_RESERVATION_RULE_SETS	Audit logging - audit ACS reservation rule sets inserts
1096	CC_CODE_TABLES_UPDATE_ACS_RESERVATION_RULE_SETS	Audit logging - audit ACS reservation rule sets updates
1097	CC_CODE_TABLES_DELETE_ACS_RESERVATION_RULE_SETS	Audit logging - audit ACS reservation rule sets deletes
1098	CC_CODE_CREATE_PASS_OPERATIONS_EVENTS	Create Pass Operations Events
1099	CC_CODE_CREATE_PASS_USAGE_EVENTS	Create Pass Usage Events
1100	CC_CODE_TICKET_FULL_UPGRADE_SEARCH	Ticket Full Upgrade Search
1101	CC_CODE_PASS_PORTAL_DISPLAY_UPGRADE_CREDIT	Pass portal display upgrade credit
1102	CC_CODE_ACS_RES_RULE_SET_ATTRIBUTE_GROUP_ID	ACS Reservation Rule Set Attribute Group ID
1103	CC_CODE_ACS_RES_RULE_SET_CONSEQUENCE_ACCESS_CODE_ATTRIBUTE_GROUP_ID	ACS Reservation Rule Set Consequence Access Code Attribute Group ID
1104	CC_CODE_ACS_RES_RULE_SET_NOTIFICATION_ACCESS_CODE_ATTRIBUTE_GROUP_ID	ACS Reservation Rule Set Notification Access Code Attribute Group ID
1105	CC_CODE_ADDRESS_LOOKUP_PROVIDER	Address Lookup Provider
1106	CC_CODE_ADDRESS_LOOKUP_PROVIDER_BASE_URL	Address Lookup Provider Base URL
1107	CC_CODE_ADDRESS_LOOKUP_PROVIDER_API_KEY	Address Lookup Provider API Key
1108	CC_CODE_EMAIL_VALIDATION_PROVIDER	Email Validation Provider
1109	CC_CODE_EMAIL_VALIDATION_PROVIDER_BASE_URL	Email Validation Provider Base URL
1110	CC_CODE_EMAIL_VALIDATION_PROVIDER_API_KEY	Email Validation Provider API Key
1111	CC_CODE_PHONE_VALIDATION_PROVIDER	Phone Validation Provider
1112	CC_CODE_PHONE_VALIDATION_PROVIDER_BASE_URL	Phone Validation Provider Base URL
1113	CC_CODE_PHONE_VALIDATION_PROVIDER_API_KEY	Phone Validation Provider API Key
1114	CC_CODE_SHOW_MULTIPLE_DAY_EVENTS_BY_STARTDATE_ONLY	Show multiple day events by start date only
1115	CC_CODE_AUTO_ORDER_CLOSE	
1116	CC_CODE_HIDE_TKT_LK_UP_RFID	Determine whether to show RFID in the ticket lookup window
1117	CC_CODE_HIDE_TKT_LK_UP_RFID_HISTORY	Determine whether to show RFID tab and history in the ticket lookup window
1118	CC_CODE_REQUIRE_TICKET_LOOKUP_GUEST_NAME_EDIT_PRIVILEGE	

1119	CC_CODE_OFFLINE_NODE_HOURS	Offline Node. Monitor how long a node has remained offline
1120	CC_CODE_OFFLINE_NODE_HOURS_MAX	Offline Node. Maximum hours a node may remain offline before action is taken
1121	CC_CODE_ENABLE_ACTIVE_DIRECTORY_AUTHENTICATION	Enable Active Directory authentication
1122	CC_CODE_ACTIVE_DIRECTORY_OFFLINE_CREDENTIALS	Active Directory offline credentials
1123	CC_CODE_MAX_CONNECTION_IDLE_SECONDS	
1124	CC_CODE_CONNECTION_IDLE_POLL_INTERVAL_SECONDS	
1125	CC_CODE_LOG_IDLE_CONNECTIONS	
1126	CC_CODE_RELEASE_IDLE_CONNECTIONS	
1127	CC_CODE_PRINT_TICKET_BEFORE_RECEIPT	Print ticket before receipt
1128	CC_CODE_SHOW_TKT_LK_UP_SERIAL_SEQ	Determine whether to show Serial sequence number in the ticket lookup window
1129	CC_CODE_RECORD_TRANS_CONTACT_ON_CONTACT_REQ_TICKET	Record transaction contact on contact required tickets
1130	CC_CODE_ONLINE_EXCHANGE_ALLOWED_EARLIEST_DAYS_BEFORE_VISIT_DATE	The earliest days before the visit to allow online exchange
1131	CC_CODE_ACTIVE_DIRECTORY_AUTO_CREATE_USERS	Active Directory auto create users
1132	CC_CODE_ACTIVE_DIRECTORY_MAP_USER_PROFILES	Active Directory map user profiles
1136	CC_CODE_EGALAXY_HONOR_RECORD_SYSTEM_COMPANY_GC	eGalaxy - Honor Record System Company General Config
1137	CC_CODE_QOP_JOURNAL_SENDER_WAIT_TIME	Maximum number of seconds QOP should wait for Journal Sender to finish sending tickets to the database
1138	CC_CODE_TRANSFER_PACKAGE_ON_PASS_UPGRADE	Transfer original package when upgrading a pass
1139	CC_CODE_END_OF_LIFE_DATE_WINDOW_DAYS	The number of days after the end of life date when the item becomes locked
1140	CC_CODE_ENABLE_END_OF_LIFE_DATE_LOCK	Indicates if the item will become locked when the end of life date window passes
1141	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_END_OF_LIFE_DATE	Ticket Lookup - option is true when End of Life Date Value is NOT selected
1142	CC_CODE_HIDE_TKT_LK_UP_ACCOUNT_END_OF_LIFE_LOCK	Ticket Lookup - option is true when End of Life Lock Value is NOT selected
1143	CC_CODE_PICK_PASS_SHOW_URGENT_NOTES	Show the urgent notes column
1144	CC_CODE_DISABLE_WINCOR_NIXDORF_MSR	Disable the Wincor-Nixdorf MSR with "c1" start sentinel
1145	CC_CODE_SALES_CHANNEL_MAX_IMAGE_SIZE	Max image upload size in kilobytes for Sales Channel images
1146	CC_ENABLE_ORCA_SMARTCARD_FEATURE	Show the tab in Services that allows configuration of ORCA Smart Card
1147	CC_CODE_ORCA_SMARTCARD_HOTLISTPOLLING_INTERVAL	Hotlist Service polling interval. The amount of time (seconds) between checks
1148	CC_CODE_TRANSITACCOUNTS_REQUEST_BASEURL	Transit Accounts Request URL
1149	CC_CODE_POSTPAYMENT_REQUEST_BASEURL	Post Payment Request URL
1150	CC_CODE_ORCA_SMARTCARD_USERNAME	Account ID
1151	CC_CODE_ORCA_SMARTCARD_PASSWORD	Account password
1152	CC_CODE_ORCA_SMARTCARD_TIMEOUT	Timeout in (seconds). Time after which a response is considered incomplete
1153	CC_CODE_ORDER_SEARCH_RESULTS_WARNING_THRESHOLD	Max number of order search results before warning is displayed
1154	CC_CODE_ORDER_SEARCH_AND_DISPLAY_PAYMENT_STATE	Search by payment state and display payment state in order
1155	CC_CODE_DOWNLOAD_FULL_TABLE_ON_FAILURE	Download the full table when a row change download fails
1156	CC_CODE_ENABLE_ORCA_SMARTCARD_UPDATE	Enable functionality to add products to a NGOrca smartcard
1157	CC_CODE_RETAIN_MEDIA_ON_UPGRADE	Retain media on upgrades (tickets only)
1158	CC_CODE_ORCA_OPERATOR_ID	ID of the public transport agency which is associated with NGOrca transactions. Assigned by INIT
1159	CC_CODE_ORCA_SALESCCHANNEL_ID	ID of the sales channel which is associated with NGOrca transactions. Assigned by INIT
1160	CC_CODE_OE_ITEM_SELECTION_PRICE_SCHEDULE_PRICING_MODE	OE Item Selection Price Schedule Pricing Mode
1161	CC_CODE_ORCA_MAX_TRANSACTION_ROWS	Maximum number of NGOrca transactions inquiry will return during a transaction inquiry
1162	CC_CODE_ORCA_EPURSE_EXTERNAL_ID	External Identifier of the NGOrca ePurse reload product
1163	CC_CODE_TABLES_CENTRAL_ORCA_CARD_PUBLIC_KEYS	Indicates if central Orca card public keys are used
1164	CC_CODE_ORCA_MIN_RELOAD_AMOUNT	Minimum epurse top up amount that can be loaded onto a ngOrca card
1165	CC_CODE_TABLES_CENTRAL_ORCA_BLOCKED_CARDS	Indicates if central Orca blocked cards are used
1166	CC_CODE_ORCA_HOTLIST_REQUEST_BASEURL	Orca hotlist request URL
1167	CC_CODE_ORCA_HOST_OFFLINE_FOP	FOP used for ngOrca card transactions when the host is offline
1168	CC_CODE_TABLES_CENTRAL_ORCA_RESULT_TYPES	Orca usage responses with override options
1169	CC_CODE_ALLOW_PAYMENTS_ON_ORDERS_WITH_INVALID_SALES_PROGRAMS	
1170	CC_CODE_PREVENT_ISSUANCE_ON_ORDERS_WITH_INVALID_SALES_PROGRAMS	
1171	CC_CODE_USE_BILL_TO_CONTACT_FOR_CREDIT_CHECK	
1172	CC_CODE_TABLES_CENTRAL_RFCS_ITEMS	
1173	CC_CODE_TABLES_CENTRAL_RFCS_OPERATIONS	
1174	CC_CODE_CONTACT_STATE_FIELD_REQUIRED	State field required or optional
1175	CC_CODE_MP_RETURN_VARIANCE_FOP	Multi-payment return upgrade variance FOP
1176	CC_CODE_MP_RETURN_REMAINDER_FOP	Multi-payment return remainder FOP
1177	CC_CODE_EXTEND_LOCKOUTS_TO_POS	Extend ACS Lockouts for sales related functions in POS/OE
1178	CC_CODE_ORDER_LINE_DETAIL_STATUS_DISPLAY_MODE	Order Line Detail Status Display Mode
1179	CC_CODE_ORCA_DEBTOR_NUMBER	Static value associated with NGOrca FarePayment transactions. Defined by ORCA
1180	CC_CODE_ORCA_CLIENT_ID	Client ID value associated with NGOrca Fare Payment transactions. Assigned by INIT
1181	CC_CODE_ENABLE_SUSPENDED_TRANSACTIONS	Enable Suspended Transactions
1182	CC_CODE_ENABLE_IBAN_ENCRYPTION	
1183	CC_CODE_TABLES_CENTRAL_DEBIT_TYPES	
1184	CC_CODE_SHOW_DESCRIPTION_ONLY_FOR_GIFT_APPEAL_SELECTION	Show description only for appeal selection in drop down list.
1189	CC_CODE_WEB_ORDER_PROCESSOR_FILENAME_PREFIX	Web Order Processor - PDF Ticket Filename Prefix.
1190	CC_CODE_EGALAXY_USE_CENTRAL_CREDIT_FOP	eGalaxy - Use central credit FOP

3 Scope values

Value	Constant	Description

0		Global scope - applies to all nodes, unless scope 1 or 2 is found for this same option
1		Agency scope - applies to all nodes in the specified agency (as defined by OwnerID), unless scope 2 is found for the same option
2		Node scope - applies to only this node (as defined by OwnerID).

### 3.29 CountyCodes

The Federal Information Processing Standard is a five-digit number assigned to each county in the U.S. by the Bureau of Census. The first two digits represent the state and the other three are the county number.

#### Columns

Column	Type	Allow Nulls	Description
CountyCodeID	Int	N	Primary key, always unique. System generated.
CountyCode	Char(6)	N	The code for the county.
Name	Char(26)	Y	The name of the county.
State	VarChar(40)	Y	The state the county is in.

#### Indexes

Name	Kind	Columns	Purpose
PKCountyCodesCountyCodeID	P	CountyCodeID	Primary key.
IXCountyCodesCountyCode		CountyCode	

### 3.30 Coupons

A ticket set consists of one or more coupons. A coupon is the smallest unit of stock that a ticket printer can process individually. The layout for each coupon is user-specified and then combined into a ticket set to be assigned to a function key within a product.

#### Columns

Column	Type	Allow Nulls	Description
CouponID	Int	N	Primary key, always unique. System generated.
Coupon	Int	N	User definable coupon number. This is the value used to identify a coupon throughout the system.
Line	Int	N	Line number within the coupon the Text field is located.
Text	varchar(99)	Y	One line of text in the coupon format
Description	varchar(40)	Y	Coupon description

#### Indexes

Name	Kind	Columns	Purpose
PKCouponsCouponID	P	CouponID	Primary key.

### 3.31 CreditCardUses

This table is designed to store all credit card use to aid in fraud prevention. Each Kiosk, at the time of payment, checks this file for number of times the card has been used in a given time period.

#### Columns

Column	Type	Allow Nulls	Description
CreditCardUseID	Int	N	Primary key, always unique
CreditCard	Char(24)	N	Creditcard number
UseDateTime	DateTime	N	Date and time of credit card use
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the credit card number (CreditCard).

#### Indexes

Name	Kind	Columns	Purpose
PKCreditCardUseID	P	CreditCardUseID	Primary Key.
IXCreditCardUseCreditCard	I	CreditCard	Adds ability to query data by creditcard

### 3.32 Currencies

This table is designed to store currency information for the system based on ISO 4217 standard currency data.

#### Columns

Column	Type	Allow Nulls	Description
CurrencyID	Int	N	Primary key, always unique
Description	VarChar(50)	N	Description of this currency
CurrencyCode	Char(3)	N	3-character currency code based on ISO 4217 standard
CurrencyCodeID	Int	N	Numeric currency code based on ISO 4217 standard
DecimalOffset	Int	N	Number of decimal places used for this currency
CurrencyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PK_CurrencyID	P	CurrencyID	Primary Key
IXCurrenciesCurrencyCodeID	I	CurrencyCodeID	Unique, ensures no duplicate CurrencyCodeID entries

### 3.33 CurrentLogons

This table stores information about the user currently logged on to a node or Manager's Workstation. It is used to determine if a user is already logged on to another station and allow prevention of concurrent logons by the same user.

#### Columns

Column	Type	Allow Nulls	Description
CurrentLogonID	Identity	N	Primary key, always unique
UserID	Integer	N	User ID of the logged on user
NodeID	Integer	Y	Node number of the node that is logged on. This will be 0 or null for Manager's workstation since it has no node number.
MachineName	nvarchar(100)	Y	Computer name from the computer that a Manager's Workstation is logged onto. This will be blank or null for logons from stations with a node number (Galaxy, ACS32)

#### Indexes

Name	Kind	Columns	Purpose
PKCurrentLogons	P	CurrentLogonID	Primary Key
IXCurrentLogonsUserID	IX	UserID	Allows selection of current logons given a user ID. Used to determine if the user is already logged on somewhere.
IXCurrentLogonsNodeID	IX	NodeID	Allows selection of current logon for a node. Used to load row for deletion.
IXCurrentLogonsNodeID	IX	MachineName	Allows selection of current logon for an MWS machine. Used to load row for deletion.

### 3.34 DailyMenus

This table contains a record of the default daily menus as configured by user.

#### Columns

Column	Type	Allow Nulls	Description
DailyMenuUniqueID	Integer	No	Primary Key always unique
DailyMenuID	Integer	No	Local key
MenuListName	Varchar(40)	Yes	'Default' for Default Daily Menus
SundayMenu	Varchar(24)	Yes	MenuID for default menu for Sundays
MondayMenu	Varchar(24)	Yes	MenuID for default menu for Mondays
TuesdayMenu	Varchar(24)	Yes	MenuID for default menu for Tuesdays
WednesdayMenu	Varchar(24)	Yes	MenuID for default menu for Wednesdays
ThursdayMenu	Varchar(24)	Yes	MenuID for default menu for Thursdays
FridayMenu	Varchar(24)	Yes	MenuID for default menu for Fridays
SaturdayMenu	Varchar(24)	Yes	MenuID for default menu for Saturdays

#### Indexes

Name	Kind	Columns	Purpose
PKDailyMenuUniqueID	P	DailyMenuUniqueID	Primary key.

### 3.35 DateSpecificCapacityExcludedNodes

This table contains all nodes that should be excluded from the Date Specific Capacity System.

#### Indexes and Constraints

Primary Key:

(None)

Indexes:

(None)

Column	Type	Allow Nulls	Description
DateSpecExNodesID	Int	N	Always unique
NodeNumber	Int	Y	Node number to exclude

### 3.36 Denominations

This table contains the definitions for currency denominations.

#### Columns

Column	Type	Allow Nulls	Description
DenominationID	Int	N	Primary key, always unique
DenomID	Int	N	Denomination ID used by the system
DenomName	Char(20)	N	Full description of the denomination
Abbreviation	Char(7)	N	Abbreviated name of the denomination
Multiplier	Float	N	Multiplier x BaseCurrency unit = Denomination (eg. \$10 has Multiplier = 10)
Decimals	Int	N	Number of decimals to use for display
CurrencyKey	Char(1)	N	Value ("A" thru "Z") indicating the key used as a shortcut to select the currency being used.
Depositable	Bit	N	Flag to indicate if this denomination can be part of a deposit
OrderedAsChange	Bit	N	Flag to indicate if this denomination can be ordered as change
DepositSign	Int	N	Value indicating whether deposit amounts of this denomination can be positive, negative, or either positive or negative. <sup>1</sup>
FOP	Int	Y	FOP associated with the denomination

#### Indexes

Name	Kind	Columns	Purpose
PKDenominationsDenominationID	P	DenominationID	Primary Key.

#### <sup>1</sup> Deposit Sign Values

Value	Description
1	Denomination can be either positive or negative
2	Denomination must be positive
3	Denomination must be negative

### 3.37 Deposits

The **Deposits** table contains the DepositIDs used by the users during the deposits and cashouts. This table is used to validate the uniqueness of the DepositID provided by the user during the deposit and cashout.

#### Columns

Column	Type	Allow Nulls	Description
DepositID	Int	N	Primary key, always unique.
UserDepositID	Char(16)	N	Deposit ID provided by a user during a Deposit.
NodeNo	Int	N	POS Node number where Deposit was done
TranNo	Int	N	Transaction number of the deposit
UserID	Int	N	ID of the user who did the Deposit.
DepositDate	DateTime	N	Date & time of the Deposit

#### Indexes

Name	Kind	Columns	Purpose
PKDepositsDepositID	P	DepositID	Primary Key.
IXDepositsUserDepositID		UserDepositID	System queries the Deposits table to determine whether the user provided DepositID is unique or not. This index is created to speed up the query.

- Note: The DepositDate might be one second off from the actual journal transaction date and time. This is due to the time differences between the time of SQL Deposits table insert and the journalization of the deposit by the system.

### 3.38 Disbursements

This table stores Galaxy's disbursement information. Each disbursement is broken up into a header and several details. Detail records can be referenced by DisbursementID. See the below DisbursementDetails table.

Column	Type	Allow Nulls	Description
DisbursementUniqueID	Int	N	Primary key, always unique. System generated.
DisbursementID	Int	N	User definable disbursement number. This is the value used to identify a disbursement throughout the system.
Name	Char(20)	Y	The user definable name of the disbursement
Inactive	Bit	N	True if disbursement is Inactive, and not visible on most picklists.
DisbursementGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKDisbursementsDisbursUniqueID	P	DisbursementUniqueID	Primary Key.
IXDisbursementsDisbursementID		DisbursementID	Unique index to enforce that a DisbursementID value is not duplicated.

### 3.39 DisbursementDetails

This table stores the detail records for the Disbursements table. Each disbursement is broken up into a header and several details. Detail records can be referenced by DisbursementID.

Column	Type	Allow Nulls	Description
DisbursementDetailID	Int	N	Primary key, always unique. System generated.
DisbursementID	Int	N	ID number of the disbursement header record. Rows in this table with the same DisbursementID value are all rows for the same Disbursement header.
Name	Char(20)	Y	The name of the disbursement detail.
Price	Float	Y	Price
Basis	Char(2)	Y	Determines the basis of the Price field: percent, amount, or remainder. <sup>1</sup>
StockType	Int	Y	Stock type used for the disbursement detail.
PrinterNo	Int	Y	The printer number used for the disbursement detail.
TicketSet	Int	Y	The ticket set used for the disbursement detail.
Company	Int	Y	The ID number of the company receiving revenue for this portion of the issued ticket.
Category	Int	Y	The transaction account category for the Company.
SubCategory	Int	Y	The transaction account sub-category for the Company
SuppressSerial	Bit	Y	Indicates if a serial will be generated. 0 means a serial is generated, 1 means serial is suppressed.
AccountNo	Char(12)	Y	User code
TaxFlag	Char(8)	Y	Bit mask for taxes. The bit is set if taxable for tax 1-8.
FkeyFlag	Char(8)	Y	Various settings for the disbursement detail. 'Y' as the first character indicates tax included.
TaxMethods	Char(8)	Y	The tax methods is an 8 character string. These eight characters refer to the eight possible taxes.
Kind	Int	Y	Disbursement detail kind. <sup>2</sup>
Value	Float	Y	*
AccessCode	Int	Y	Access code
SequenceNo	Int	Y	The index number of the disbursement part. It will be a value between 1 and 10.
UseTaxTable	Bit	Y	Indicates if tax tables are to be used for this item
TaxTableID	Int	Y	Value of the tax table header for this item
TaxTableMethod	Int	Y	Indicates if tax is per item(0), or transactional (1)
DisbursementDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

- Each of the noted columns correspond to a field on the Items table. For column descriptions, see the Items table.

#### Indexes

Name	Kind	Columns	Purpose
PKDisbursementsDetailID	P	DisbursementDetailID	Primary Key.
IXDisbursementDetailsDisbID		DisbursementID	Not currently used by the system, but one day can be used to gather all the details for a given header record.

#### <sup>1</sup> Basis Values

Value	Description
%	Percentage
\$	Actual amount
R	Remainder

#### <sup>2</sup> Kind Values

Value	Const Name	Description
10	DEBIT_FKEY	Debit Card
11	RECHARGE_FKEY	Recharge
12	REISSUE_DEBIT_FKEY	Debit Reissue
13	UPGRADE_PASS_FKEY	Upgrade Pass

### 3.40 DiscountPrivileges

This table is used to link a discount to a UserProfile.

#### Columns

Column	Type	Allow Nulls	Description
DiscountPrivilegeID	Int	N	Primary key, always unique
DiscountID	Int	Y	Reference to Discount table
UserProfileID	VarChar(10)	Y	Reference to UserProfile table
DiscountPrivilegeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKDiscountPrivilegeID	P	DiscountPrivilegeID	Primary Key

### 3.41 Discounts

This table stores Galaxy's Discount information.

#### Columns

Column	Type	Allow Nulls	Description
DiscountUniqueID	Int	N	Primary key, always unique
AccessDate	Datetime	Y	Not used
AccessUser	Int	Y	Not used
Marked	Bit	Y	Not used
DiscountID	Int	Y	Unique user-defined discount ID.
Abbr	Char(10)	Y	Abbreviation for the discount used for reports.
Name	Char(25)	Y	Name to identify the discount.
DiscType	Int	Y	Not used
ValidFrom	DateTime	Y	The starting date from which this discount may be used. Null if there is no start date.
ValidThru	DateTime	Y	The final date on which this discount may be used. Null if there is no end date.
TktSet	Int	Y	Ticket set to print when the discount is applied.
Printer	Int	Y	Printer number to print the ticket set on.
SalesDays	Int	Y	8-bit bit-mask indicating which days the discount can be used. Also determines if the discount can be used on holidays. If the bit is 1, the discount may not be used. Bits 1-7 (from right to left) represent Sunday through Monday. Bit 8 represents holidays.
AllowMulti	Bit	Y	If 1, the discount may be applied multiple times in a transaction.
SalesPerson	Char(4)	Y	User ID of the salesperson assigned to this discount.
PromoRepSuppress	Bit	Y	If 1, this discount is not included in the Promotions report.
MaxApplied	Int	Y	The maximum number of times this discount can be applied in a transaction.
OrDiscs	Bit	Y	If 1, only one of the discount details will be applied. If 0, all discount details may be applied.
OrRequirements	Bit	Y	If 1, the discount can be applied if any of the requirements are met. If 0, all requirements must be met to apply the discount.
PassRequired	Char(1)	Y	Indicates if the discount is pass required. <sup>1</sup>
PassIDStartPos	Int	Y	The start position of the pass ID in the discount barcode. (Not used)
PassIDLength	Int	Y	The length of the pass ID in the discount barcode. (Not used)
StoreBarcode	Bit	Y	If 1, the discount barcode is recorded in a memo record in the journal.
PreventUserSelection	Bit	Y	If corresponding option is selected within local DISCOUNT.DAT file, Galaxy will not allow the discount to be selected from the Discounts picklist (the discount can still be applied if selling a ticket or item with the discount attached).
ExternalDiscountID	VarChar(20)	Y	Alphanumeric ID of the discount, user can enter any alpha numeric characters which is be used to identify the Discount in the web store
NoteID	Int	Y	Foreign Key, references Notes SQL table. The note is used to store instructions to use the discount.
ApprovalRequired	Bit	Y	Determines if profile approval requirements are enabled
DiscountCode	nvarchar(30)	Y	A unique alphanumeric code for this discount.
AttributeValueGroupID	Int	Y	Links Discounts to the AttributeValues table
Inactive	Bit	Y	True if Discount is Inactive, and not visible in most picklists.
AllowDiscountOtherRequirements	Bit	Y	If 1, allows this discount to be applied to an item that is a requirement for a different discount. It also allows requirements for this discount to be discounted by other discounts.
DiscountGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### <sup>1</sup> PassRequired Values

Value	Description
N	Not pass required.
Y	The discount is pass required and the pass can be used once per day.
M	The discount is pass required, and the pass can be used multiple times in a day.

#### Indexes

Name	Kind	Columns	Purpose
PKDiscountsDiscountUniqueID	P	DiscountUniqueID	Primary Key
IXDiscountsDiscountID	IX	DiscountID	Index to increase search speed
IXDiscountsDiscountCode	IX	DiscountCode	Unique index on DiscountCode

### 3.42 DiscountDetails

This table stores the discount item list.

#### Columns

Column	Type	Allow Nulls	Description
DiscountDetailID	Int	N	Primary key, always unique
DiscountID	Int	N	Foreign key to Discounts table
DiscountGroup	NVarChar(10)	Y	The discount group to which the product must belong to allow the discount to be applied.
Company	Int	Y	The ID of the company required on the product to apply the discount. If 0, any company may apply the discount.
Category	Int	Y	The ID of the account category needed to apply the discount. If 0, any category may apply the discount.
SubCategory	Int	Y	The ID of the account subcategory needed to apply the discount. If 0, any subcategory may apply the discount.
Qty	Int	Y	The number of products in a transaction to which this discount detail may be applied.
Amount	Money	Y	The amount or percentage of the normal price that will be discounted.
Basis	Char(1)	Y	If %, the amount is a percentage off the normal price.
Printer	Int	Y	The printer for the ticket set that prints when this discount detail is applied.
TicketSet	Int	Y	The ticket set that prints when this discount detail is applied.
Priority	Char(1)	Y	Not used.
ReplacementCompany	Int	Y	When non-zero, the company ID to use for the product when this discount is applied.
ReplacementCategory	Int	Y	When non-zero, the account category ID to use for the product when this discount is applied.
ReplacementSubCategory	Int	Y	When non-zero, the account subcategory ID to use for the product when this discount is applied.
ReplacementPrinter	Int	Y	When non-zero, the printer to use to print the ticket set for the product when this discount is applied.
ReplacementTicketSet	Int	Y	When non-zero, the ticket set to use for the product when this discount is applied.
ReplacementDisbursementID	Int	Y	When non-zero, the disbursement ID to use to disburse the product to which this discount is applied.
CodeType	Int	Y	The type of item needed for this detail (by journal code). <sup>1</sup>
Unlimited	Bit	Y	If 1, this detail can be applied to an unlimited number of products in a transaction.
UseTaxTable	Bit	Y	Indicates whether to use tax table for the item
TaxTableID	Int	Y	Value of the tax table header for this item
TaxTableMethod	Int	Y	Indicates if tax is per item(0), or transactional (1)
ReplacementAccessCode	Int	Y	The Access Code used for the replacement ticket, if that gets activated when the discount is applied
SequenceNo	Int	Y	Value from 1 to 10 indicating the sequence of this discount detail
AdjustMembershipExpiration	Bit	Y	Determines if the discount, when applied to a Membership, will adjust the expiration date when applied
AdjustMembershipExpirationType	Char(1)	Y	The type of expiration adjustment that will be applied with the discount. <sup>2</sup>
AdjustExpirationDaysQty	Int	Y	The number of days to extend the expiration date of the Membership by
AdjustExpirationMonthsQty	Int	Y	The number of months to extend the expiration date of the Membership by
AdjustExpirationToEndOfMonth	Bit	Y	If extending the membership by a number of months, setting this to Y will also extend the Membership until the end of the new month
AdjustExpirationSpecificDate	Datetime	Y	The date to extend the expiration of the Membership to
PLU	NVarChar(20)	Y	PLU of the item to apply a discount to when restricting the discount by a single product.
ItemGroupID	Int	Y	Unique ID of the item group to which the product must belong for the discount to be applied. May be used when restricting the discount by multiple products.
RestrictionType	Int	Y	Method of restricting the products to which the discount may be applied. <sup>2</sup>
DiscountDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKDiscountDetailsID	P	DiscountDetailID	Primary Key
IXDiscountDetailsDiscID	FK	DiscountID	To be able to select all the details for a given discount

#### <sup>1</sup> CodeType Values

Value	Description
0	Ticket (code 101)
1	Item (code 102)
255	Either tickets or items (101 or 102)

#### <sup>2</sup> AdjustMembershipExpirationType Values

Value	Description
D	Extend the expiration by a number of days as indicated by the AdjustExpirationDaysQty field
M	Extend the expiration by a number of months as indicated by the AdjustExpirationMonthsQty field
S	Extend the expiration of the membership to a specific date as indicated by the AdjustExpirationSpecificDate field

#### <sup>2</sup> RestrictionType Values

Value	Description
0	Discount restriction by multiple products.
1	Discount restriction by a single product.

### 3.43 DiscountRequirements

This table stores the requirements in order to receive discount from the Discounts table.

#### Columns

Column	Type	Allow Nulls	Description
DiscountRequirementID	Int	N	Primary key, always unique
DiscountID	Int	N	Foreign key to Discounts table
DiscountGroup	NVarChar(10)	Y	The discount group to which the product must belong to meet the requirement.
Company	Int	Y	The ID of the company required on the product to meet the discount. If 0, any company meets the requirement.
Category	Int	Y	The ID of the account category needed to meet the requirement. If 0, any category meets the requirement.
SubCategory	Int	Y	The ID of the account subcategory needed to meet the requirement. If 0, any subcategory meets the requirement.
Qty	Int	Y	The quantity needed to meet the requirement.
AllowDiscount	Bit	Y	If 1, a discount may be applied to the requirement.
CodeType	Int	Y	The type of item needed for this requirement (by journal code). <sup>1</sup>
SequenceNo	Int	Y	Value from 1 to 10 indicating the sequence of this discount requirement
PLU	NVarChar(20)	Y	PLU of the item to be used as a discount requirement when restricting the requirement by a single product.
ItemGroupID	Int	Y	Unique ID of the item group to which the product must belong for the discount to be used as a requirement. May be used when restricting the requirement by multiple products.
RestrictionType	Int	Y	Method of restricting the products for which the discount may be used as a requirement. <sup>1</sup>
DiscountRequirementGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### <sup>1</sup> CodeType Values

Value	Description
0	Ticket (code 101)
1	Item (code 102)
255	Either tickets or items (101 or 102)

#### Indexes

Name	Kind	Columns	Purpose
PKDiscountRequirementID	P	DiscountRequirementID	Primary Key.
IXDiscountReqDiscID	FK	DiscountID	To be able to select all the requirements for a given discount

#### <sup>1</sup> RestrictionType Values

Value	Description
0	Discount restriction by multiple products.
1	Discount restriction by a single product.

### 3.44 DiscountValidations

The zip code ranges applicable to use a Discount can be defined here. The table has column DiscountID which references to the Discounts table.

#### Columns

Column	Type	Allow Nulls	Description
DiscountValidationID	Int	N	Primary key, always unique
Name	VarChar(80)	N	Name of the discount validation
DiscountID	Int	N	FK reference to Discounts table
StartRange	VarChar(30)	Y	Start data value to validate if the validation is a range (i.e. Zip Code range)
EndRange	VarChar(30)	Y	End data value to validate if the validation is a range
ValidationData	VarChar(30)	Y	Data to validate if the validation is not a range (i.e. State)
DiscountValidationGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKDiscountValidationID	P	DiscountValidationID	Primary Key.

### 3.45 DMACodes

This table used in the maintenance of Zip Code area Demographic Marketing Codes.

#### Columns

Column	Type	Allow Nulls	Description
DMACodeID	Int	N	Primary key, always unique
Code	Varchar(3)	N	Local key
Name	Varchar(26)	Y	Optional name

#### Indexes

Name	Kind	Columns	Purpose
PKDMACodeID	P	DMACodeID	Primary Key.

### 3.46 DuplicateReferences

This table contains duplicate references from surveys that utilize references.

#### Columns

Column	Type	Allow Nulls	Description
DuplicateReferenceID	Int	N	Primary key, always unique.
UserName	NVarChar(24)	Y	The username of the current user logged on.
NodeID	Int	Y	The node number of the current station.
DuplicateDate	DateTime	Y	Date/time of the duplicate reference.
ReasonID	Int	Y	Reason ID used for the reference.
Reference	NVarChar(255)	Y	The reference string that was duplicated.

#### Indexes

Name	Kind	Columns	Purpose
PKDuplicateReferenceID	P	DuplicateReferenceID	Primary key

### 3.47 ExchangeDetails

The ExchangeDetails table is a detail table associated to the ExchangeMethods table.

#### Columns

Column	Type	Allow Nulls	Description
ExchangeDetailID	Int	N	Primary key, always unique
ExchangeID	Int	Y	FK to ExchangeMethods.ExchangeID
Scale	Int	Y	Number of times the ticket was used OR number of days left for the pass, depending on the value of ExchangeMethods.BasedOn.
Kind	Char(1)	Y	Determines the kind for this detail <sup>1</sup>
Value	Money	Y	Percentage when Kind = 'T' or amount when Kind = 'A'.
PLU	Char(20)	Y	PLU when Kind = 'P'

#### Indexes

Name	Kind	Columns	Purpose
PKExchangeDetailID	P	ExchangeDetailID	Primary Key

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
'P'	KIND_DEDUCT_PLU	Item PLU
'T'	KIND_DEDUCT_PERCENT	Connect method of file
'A'	KIND_DEDUCT_AMOUNT	Amount

### 3.48 ExchangeMethods

The ExchangeMethods table allows you to define ways to calculate prorated returns of passes.

#### Columns

Column	Type	Allow Nulls	Description
ExchangeMethodID	Int	N	Primary key, always unique
ExchangeID	Int	Y	Unique auto-incrementing value
Code	Char(5)	Y	The code for this exchange method
Description	VarChar(40)	Y	The description for this exchange method
BasedOn	Char(1)	Y	The basis for this exchange method <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKExchangeMethodID	P	ExchangeMethodID	Primary Key

<sup>1</sup> BasedOn Values

Value	Gateway Constant Name	Description
'U'	EXCHANGE_BASED_ON_USAGE	Number of usages
'D'	EXCHANGE_BASED_ON_DAYS	Number of days (lower limit)
'P'	EXCHANGE_BASED_ON_PER_USE	Number of usages

### 3.49 ExternalCalls

This table stores information about external programs called by Galaxy.

#### Columns

Column	Type	Allow Nulls	Description
ExternalCallID	Int	N	Primary key, always unique.
Description	Char(20)	Y	Description shown on POS buttons
ProgramPath	Varchar(256)	Y	Path and filename of External Call
ProgramParams	Varchar(256)	Y	Optional command line parameters for External Call

#### Indexes

Name	Kind	Columns	Purpose
PKExternalCallsExternalCallID	P	ExternalCallID	Primary Key.

### 3.50 ExternalConnections

This table stores connection information to the different external hosts. Currently we store the URL of the web sites where we send the data or the IP address of the FTP site where we upload the messages.

Columns

Column	Type	Allow Nulls	Description
ExternalConnectionID	Int	N	Primary key, always unique
Name	VarChar(80)	N	Name of the connection
Host	VarChar(256)	Y	When ExternalConnections.Method is 0 (HTTP/POST) or 3 (HTTP/PUT) then ExternalConnections.Host field contains a URL. When ExternalConnections.Method is 1 (FILE) then ExternalConnections.Host field contains a file path. When ExternalConnections.Method is 2 (FTP) then ExternalConnections.Host field contains an IP address.
UserName	VarChar(128)	Y	Username to use when connecting to the host
Password	VarChar(128)	Y	Password to use when connecting to the host
Method <sup>1</sup>	Int	Y	Method of connection <sup>1</sup>
FTPPassiveMode	Bit	Y	True if FTP Passive mode will be used.
AuthenticationMethod	Int	Y	Method of passing credentials to host for HTTP. <sup>2</sup>
ApplyCount	Int	Y	Indicates the "version" of the database that has been published to this external connection. This is used by the publishing process when publishing to this external connection. When a publish operation to this connection completes successfully, ApplyCount is updated to the current PublishCount. If ApplyCount is NULL or 0, a publish operation has never been completed. This is analogous to the ApplyCount column in NodeDataStatus.
ClientID	NVarChar(128)	Y	Client ID value to use when connecting to the host using OAUTH2
ClientSecret	NVarChar(128)	Y	ClientSecret to use when connecting to the host using OAUTH2
ClientAuthURL	VarChar(256)	Y	Authorization URL to use when connecting to the host using OAUTH2

Indexes

Name	Kind	Columns	Purpose
PKExternalConnectionID	P	ExternalConnectionID	Primary Key.

#### <sup>1</sup> Method Values

Value	Gateway Constant Name	Description
0	HTTP_EXTERNAL_CONNECT_METHOD	Connect method of HTTP/Post
1	FILE_EXTERNAL_CONNECT_METHOD	Connect method of file
2	FTP_EXTERNAL_CONNECT_METHOD	Connect method of FTP
3	HTTP_PUT_EXTERNAL_CONNECT_METHOD	Connect method of HTTP/Put

#### <sup>2</sup> AuthenticationMethod Values

Value	Gateway Constant Name	Description
0	CLEARTEXT_EXTERNALCONNECTION_AUTHENTICATIONMETHOD	Send credentials in clear text
1	BASIC_EXTERNALCONNECTION_AUTHENTICATIONMETHOD	Send credentials using basic authentication

### 3.51 ExternalFieldConnections

This table contains field connection IDs between Galaxy and External Data. This table is used to link Galaxy data at the row level to external data.

#### Columns

Column	Type	Allow Nulls	Description
ExternalFieldConnectionID	Int	N	Primary key, always unique
TableID	Int	N	Internal Table ID used to identify the field connection data <sup>1</sup>
ConnectionType	Int	N	This defines the external field connection type <sup>2</sup>
InternalID	Int	N	Gateway ID (mostly the Primary Key ID from Gateway Counters table, for FOP table this is the FOPCode)
ExternalID	Int	N	ID used by External System to uniquely identify the row.
Status	Int	Y	Holds the status of the connection whether it is active or inactive. Currently only CustContacts marked inactive are not merged into Raiser's Edge. <sup>3</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKExternalFieldConnectionID	P	ExternalFieldConnectionID	Primary Key.
IXExtFldConTblIDConTypeExtID		TableID, ConnectionType, ExternalID	Enhance the speed of the SelectByExternalID query.
IXExtFldConTblIDConTypeIntID		TableID, ConnectionType, InternalID	Enhance the speed of the SelectByInternalID query.

<sup>1</sup> Table ID Values

TableID	Table
14	PASSES
110	ADDRESS
113	CUSTCONTACTS
141	RELATIONSHIP TYPES
142	RELATIONSHIPS
143	NAME TITLES
144	NAME SUFFIXES
145	CAMPAIGNS
146	FUNDS
147	APPEALS
148	SOLICITATIONS
149	GIFTS
150	GIFT DETAILS
151	PASS HISTORY
153	USAGE
166	SUPER TICKETS
284	PKG INSTANCE DETAILS
340	JOINT MEMBERS

<sup>2</sup> ConnectionType Values

Value	Const Name	Description
1	CT_RAISERS_EDGE	The field connection is for The Raiser's Edge

<sup>3</sup> Status Constants

Value	Gateway Constant Name	Description
0	RE_CONNECTION_STATUS_ACTIVE	Connection is active and should be merged (Default Value)
1	RE_CONNECTION_STATUS_INACTIVE	Connection is active and should not be merged (Only for CustContacts currently).

### 3.52 ExternalSiteRecords

This table is used to hold the VisualID and the name of the target table for incoming tickets and passes from external sites. Currently, the insert and update triggers on the passes and tickets table use this table to determine if GxTrigger records are to be created. If the ticket or pass being saved is in this table, GxTrigger records should not be created to be distributed to other sites.

#### Columns

Column	Type	Allow Nulls	Description
ExternalSiteRecordID	Int	N	Primary key, always unique
VisualID	varchar(40)	N	VisualID of the incoming ticket or pass
TableName	varchar(30)	N	Name of the table that holds a record with this visualID. Currently Tickets or Passes

#### Indexes

Name	Kind	Columns	Purpose
PKExternalSiteRecordID	P	ExternalSiteRecordID	Primary Key

### 3.53 ExternalVariables

This table contains definitions and values of external variables that can be used to influence various scripts throughout the system.

#### Columns

Column	Type	Allow Nulls	Description
ExternalVariableID	Int	N	Primary key, always unique
Name	varchar(100)	N	The name of the variable. This must start with a character and only contain alpha numeric characters or underscores
Description	varchar(100)	N	The description of the variable
DataType	Int	N	Indicates the data type of the variable. <sup>1</sup>
WithValue	nvarchar(max)	N	The value of the variable formated as a string based on the data type. <sup>2</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKExternalVariableID	P	ExternalVariableID	Primary Key.
IXExternalVariablesName		Name	Unique index to enforce that variable names are not duplicated.

<sup>1</sup> Table ID Values

DataType	Name
1	Boolean
2	DateTime
3	Integer
4	String
5	Float

<sup>2</sup> Data Formats

DataType	Format
1 (Boolean)	"0" for false "1" for true
2 (Integer)	Integer value formated as a string
3 (DateTime)	"yyyy-MM-ddTHH:mm:ss" (Example: "2010-08-13T09:33:57")
4 (String)	String value
5 (Float)	Float value formated as a string

### 3.54 Fares

The Fares table is used by the Transportation module and Web Store for Transportation.

#### Columns

Column	Type	Allow Nulls	Description
FareID	Integer	N	Primary Key
Carrier	Char(4)	Y	Refers to Carriers.Carrier
Origin	Integer	Y	Refers to Cities.Code Includes specific city codes and wildcards for * / STATE and * / *
Destin	Integer	Y	Refers to Cities.Code Includes specific city codes and wildcards for * / STATE and * / *
OriginZone	Integer	Y	Refers to Zones.ZoneID
DestinZone	Integer	Y	Refers to Zones.ZoneID
OW	Bit	Y	True if fare is available one way
RT	Bit	Y	True if fare is available round trip
OWFare	Smallmoney	Y	Actual fare if Method is 'V' or '\$'
RTFare	Smallmoney	Y	Actual fare if Method is 'V' or '\$'
Restrictions	Varchar(36)	Y	Semi-free form A7,S,X67H etc
TariffID	Integer	Y	Refers to Tariffs.TariffID While this is the foreign key reference for tariffs, it is also used to get honoring carriers
TemplateID	Integer	Y	Refers to Templates.TemplateID
Method	Char(1)	Y	'M' for mileage based, 'C' for combined, otherwise '\$'
MethodID	Integer	Y	Refers to MRT.MRTID
EffDate	DateTime	N	Effective date
ExpDate	DateTime	Y	Expiration date
EffSell	DateTime	Y	Date to start selling (EffDate - Adv Purchase)
ExpSell	DateTime	Y	Date to stop selling (ExpDate - Adv Purchase)
OWLimit	SmallInt	Y	Number of days one way ticket is valid
RTLimit	Smallint	Y	Number of days round trip ticket is valid

#### Indexes:

Name	Kind	Column	Purpose
PKFaresFareID	P	FareID	Primary Key
IXFaresEffExpDateOriginDestin	I	EffDate, ExpDate, Origin, Destin	Index on EffDate, ExpDate, Origin, and Destin columns
IXFaresOriginZoneDestinZone	I	OriginZone, DestinZone	Index on OriginZone and DestinZone columns
IXFaresOriginDestinEffDate	I	Origin, Destin, EffDate	Index on Origin, Destin, and EffDate columns

### 3.55 FieldAttributeGroups

This table contains header records for FieldAttributes.

Column	Data Type	Allow Null	Description
FieldAttributeGroupID	Int	N	Primary Key.
Type	VarChar(30)	Y	The name of the record type that contains attributes on the fields.
Name	VarChar(30)	Y	The meaningful name for the field attribute collection to allow for easy identification when assigning the field attribute concept to a data entry point.
Description	VarChar(256)	Y	More information on the collection of field attributes.
FieldAttributeGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKFieldAttributeGroupID	PK	FieldAttributeGroupID	Primary key

### 3.56 FieldAttributes

This table contains detail records for FieldAttributeGroups and stores information revealing what field attributes are necessary for a specific field.

Column	Data Type	Allow Null	Description
FieldAttributesID	Int	N	Primary Key.
FieldAttributeGroupID	Int	N	This column represents the set of field attributes to which the current attribute belongs. This field references the primary key of the <i>FieldAttributeGroups</i> table.
FieldName	VarChar(30)	Y	The name of the field in the record type of the FieldAttributeGroup.
Required	Bit	Y	Indicates if this field requires input from the user.
Display	Bit	Y	Indicates if this field will be displayed on the web store.
FieldAttributeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKFieldAttributeID	PK	FieldAttributeID	Primary key

### 3.57 FOPDetails

This table contains additional information about FOP. This is a SQL only information and there is no Btree implementation.

Column	Data Type	Allow Nulls	Description
FOPDetailID	Int	N	Primary Key.
FOP	Int	N	FOP this detail belongs to. This is a foreign key to FOPs.FOPCode.
UseOnWebStore	Bit	N	This option is used to indicate this FOP is used in Web Store. Also, this is used in Web Store publish to only include the FOP with this option turned on.
LongName	Varchar(50)	N	Long name string used in Web Store.
Protocol1URL	VarChar(256)	Y	URL to use for authorization of protocol 1.
Protocol2URL	VarChar(256)	Y	URL to use for authorization of protocol 2.
Protocol3URL	VarChar(256)	Y	URL to use for authorization of protocol 3.
FOPDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKFOPDetailID	PK	FOPDetailID	Primary Key
IXFOPDetailsUseOnWebStore		UseOnWebStore	Index used to retrieve all FOP
IXFOPDetailsFOP		FOP	Unique index to prevent duplicate entries

### 3.58 FOPMenuHeaders

The **FOPMenuHeaders** table is used to create custom FOP Menus that can override the standard Online Menu Configuration. The menu items are stored in the **FOPMenuDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
FOPMenuHeaderID	Int	N	Primary key, always unique. System generated.
Name	NVarChar(50)	N	Name of the FOP Menu
Description	NVarChar(250)	Y	Description of the FOP Menu
Inactive	Bit	N	True if the FOP Menu is Inactive, and not visible on most picklists.

#### Indexes

Name	Kind	Columns	Purpose
PKFopMenuHeadersFopMenuHeaderID	P	FOPMenuHeaderID	Primary key.

### 3.59 FOPMenuDetails

The **FOPMenuDetails** table is used to create custom FOP Menus that can override the standard Online Menu Configuration. The menu items are stored in the **FOPMenuDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
FopMenuDetailID	Int	N	Primary Key, Always unique.
FopMenuHeaderID	Int	N	Foreign Key to FopMenuHeaders.FopMenuHeaderID
Position	Int	N	A value, 1 to 9, that corresponds to which menu item this will be in Galaxy.
Kind	Int	N	Defines what kind of selection this menu item is. <sup>1</sup>
ItemLabel	NVarChar(30)	N	The label to display on the Menu in Galaxy
FopMenuItemID	Int	Y	The ID of the Fop Menu this detail item is assigned to. Foreign Key to FopMenuHeaders.FopMenuHeaderID
FunctionID	Int	Y	The ID of the Online Function this detail item is assigned to.
FOPCode	Int	Y	The Code of the FOP this detail item is assigned to. Foreign Key to Fops.FopCode

#### Indexes

Name	Kind	Columns	Purpose
IXFopMenuDetailsFopMenuDetailID	P	FOPMenuDetailID	Primary key.
IXFopMenuDetailsFopMenuHeaderID	F	FopMenuHeaderID	Used to select the details for a particular FopMenu.

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	MENU_KIND_MENU	TargetID refers to a FOPMenuHeaders.FOPMenuHeaderID
2	MENU_KIND_FOP	TargetID refers to a FOPs.FOPID
3	MENU_KIND_FUNCTION	TargetID refers to an Online Function

**3.60 FOPs**

The table stores the Form Of Payment information.

**Columns**

Column	Type	Allow Nulls	Description
FOPID	Int	N	Primary key, always unique
FOPCode	Int	N	Alternate Key, Form Of Payment
Name	Char(7)	N	
FOPPercent	Char(1)	N	
MPMode	Char(1)	N	Multipayment mode (Y/N)
CalcChange	Char(1)	N	Calculate change is set for this FOP (Y/N)
TktSet	Int	N	
Action	Char(1)	N	
Endorsement	Char(32)	N	
FOPMessage	Char(32)	N	
Reminder1	Char(32)	N	
Reminder2	Char(32)	N	
Query	Char(32)	N	
UseAlt	Bit	N	
EndorseEdit	Int	N	See EndorseEdit values below <sup>2</sup>
Suppress	Bit	N	
HasAccounts	Bit	N	
Printer	Int	N	
AuthNode1	Int	N	
AuthNode2	Int	N	
AuthNode3	Int	N	
Protocol1	Char(10)	N	
Protocol2	Char(10)	N	
Protocol3	Char(10)	N	
LocalAuth	Char(3)	N	
Authorize	Bit	N	
AllowDialer	Bit	N	
SubmitPreAuth	Bit	N	
PromptID	Char(4)	N	
ChangeFOP	Int	N	
ChangePrinter	Int	N	
ChangeTktSet	Int	N	
AllowCashBack	Bit	N	
AllowReferral	Int	N	
ZeroTrans	Bit	N	
AuthZeroTrans	Bit	N	
VerifyAccount	Bit	N	
IsVoucherFOP	Bit	N	
VoidPrinter	Int	N	Printer to print Void Ticket Set on
VoidTktSet	Int	N	Void Ticket Set ID
AllowMultiple	Bit	Y	
DefaultValue	Float	Y	
PromptQty	Bit	Y	
SuppressTenderPrompt	Bit	Y	
MediaID	Int	Y	
PrevSaleDate	Bit	Y	
AuthSwipedOnly	Bit	Y	
EnableCVV2	Bit	Y	
EnableCVS	Bit	Y	
DoNoMaskFOP	Bit	Y	
Currency	Char(1)	Y	Default currency to use - Blank is base currency
CaptureSignature	Bit	Y	Flag to capture signature for FOP
DoNotOpenCashDrawer	Bit	Y	Prevent drawer from opening for this FOP payment
LoadCustomer	Bit	Y	Flag to denote if the customer should be auto loaded and charge account verified.
AllowOfflineCustomer	Bit	Y	Flag to allow FOP to be used if the customer cannot be loaded
ZeroAmountRequired	Bit	Y	Set to true when the FOP can only be used if the transaction amount is \$0.00
EncryptData	bit	Y	If true, the data associated with this FOP (eg. Credit card number) will be encrypted before it is stored in the database.
CreditCardType	Int	Y	Specifies the type of credit card, if the Endorsement Edit is "Credit Card" (EndorseEdit=1) <sup>1</sup> .
SVTypeID	Int	Y	ID of the associated SV Type
VerifyManualEntry	Bit	Y	FOP config option used by CMS Points protocol
UseDynamicCurrencyConversion	Bit	Y	Determines if the FOP uses the Paymetech/NetConnect Dynamic Currency (DCC) option
DCCAdditionalDisclosure	Bit	Y	Determines if this FOP requires an additional disclosure by the purchaser when using DCC

RequireRefundReason	Bit	Y	Requires a reason to be collected for a refund payment.
PreventReturnAuth	Bit	Y	Determines if Galaxy will send return payments to the payment processor for return/refund payments. If the value is true, then return payments are automatically approved without sending a request to the processor.  Note that this option is only honored when the "Capture" flag on the FOP is also enabled. So if capture is turned off, and PreventReturnAuth is turned off, the return will still not be validated. This flag is useful when a processor needs to authorize returns for some FOPs, but not all; or if a processor needs to authorize voids, but not returns.
EnableAVSForSwiped	Bit	Y	Enable AVS For Card Present (swiped) Transactions
RemoteEndorsementMethod	Int	Y	Indicates the endorsement method that should be used by a payment plugin host if the card data is collected by the plugin instead of Galaxy. This value is sent to the plugin in the Entry Method field. The plugin may use this to determine the way the card data must be entered by the user. <sup>3</sup>
HostConfigData	NVarChar(50)	Y	Configuration data from a payment plugin host. This data is sent in authorization requests to the plugin. The format of the data is determined by the plugin.
FOPGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**Indexes**

Name	Kind	Columns	Purpose
PKFOPsFOPID	P	FOPID	Primary Key, always unique
AKFOPsFOPCode	A	FOPCode	Alternate Key, always unique

**<sup>1</sup> CreditCardType Values**

Value	Gateway Constant Name	Description
0	CREDIT_CARD_UNSPECIFIED	Credit card type is not specified, or is a type not currently listed.  This value is fine to use for many authorization protocols, even on credit card forms of payment such as the ones listed below. However, some protocols (such as Global Payments Central platform) require this information in order to format request messages properly (especially if performing functionality such as card verification that can work differently for different card types).
1	CREDIT_CARD_VISA	Visa
2	CREDIT_CARD_MC	MasterCard
3	CREDIT_CARD_AMEX	American Express
4	CREDIT_CARD_DISCOVER	Discover

**<sup>2</sup> EndorseEdit Values**

Value	Gateway Constant Name	Description
0	NO_EDIT	
1	CREDIT_CARD_EDIT	Any bankcard
2	NUMERIC_EDIT	
3	DEBIT_CARD_EDIT	
4	CHECK_EDIT	
5	EUROCHECK_CARD_EDIT	
6	CASH_CASSETTE_EDIT	
7	REISSUE_EDIT	For Transportation Reissues
8	FILLER8_EDIT	
9	FILLER9_EDIT	
10	CHARGE_ACCOUNT_EDIT	Charge account in customer file
11	FILLER11_EDIT	
12	FORMATTED_PROMPT	Formatted endorsement prompt
13	POINTS_DEBIT_EDIT	Debit points deduction
14	SCAN_EDIT	
15	STORED_VALUE_EDIT	Stored value / Rewards card
16	PREPAY_EDIT	
17	LOYALTY_EDIT	Loyalty program
18	IBAN_EDIT	For International Bank Account Number prompt.

**<sup>3</sup> RemoteEndorsementMethod Values**

Value	Gateway Constant Name	Description
0	ENDORSEMENT_BLANK	Not applicable
1	ENDORSEMENT_SWIPED	Collect the account number from a magnetic swipe reader.
2	ENDORSEMENT_MANUAL	Collect the account number from a typed-in endorsement prompt.
3	ENDORSEMENT_SCANNED	Collect the account number from a scan at an endorsement prompt.
4	ENDORSEMENT_CONTACTLESS	The account number was collected by contactless method.
5	ENDORSEMENT_CHIP	The account number was collected using an EMV chip.
6	ENDORSEMENT_FSWIPE	The account number was swiped due to a chip failure.

### 3.61 Holidays

#### Columns

Column	Type	Allow Nulls	Description
HolidayID	Int	N	Primary key, always unique
Description	Varchar(50)	Y	
BeginDateTime	DateTime	N	
EndDateTime	DateTime	Y	
HolidayGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKHolidaysHolidayID	P	HolidayID	Primary Key.
IXHolidaysBeginEndDateTime	IX	BeginDateTime, EndDateTime	This unique index matches the unique key on the BTree Holidays database. As Btree does not allow duplicates, this index will come into play down the road when editing Holidays directly from SQL is supported.

### 3.62 HypercomBatchEntries

The **HypercomBatchEntries** table is used only when authorizing credit card transactions using the Hypercom credit card protocol. It contains information needed during the settlement process (populated by Galaxy, used by Hsettle).

#### Columns

Column	Type	Allow Nulls	Description
HypercomBatchEntryID	Int	N	Primary key, always unique
Status	Int	N	Indicates the current settlement status <sup>1</sup>
Node	Int	N	The node number of the station that performed the authorization.
TransDateTime	DateTime	N	Date and time when the transaction occurred.
TransNo	Int	N	Transaction number from the journal of the authorizing node
AuthCount	Int	N	Sequence number of authorization within the transaction (normally one, but possibly higher if using multi-auth mode).
BatchID	Int	N	Batch number when posted, zero (0) if status is still pending.
BatchDateTime	DateTime	Y	Date and time when the batch entry was posted.
CardNo	VarChar(40)	N	The credit card account number that was authorized (in encrypted form)
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the credit card account number (CardNo).
CardExpr	Char(4)	N	The 4-digit credit card expiration date (MMYY format).
Amount	Money	N	The payment amount charged to the credit card.
AuthCode	VarChar(128)	N	The authorization's approval code.
HostDateTime	DateTime	N	The transaction date/time as returned from the Hypercom host.
ProcessingCode	Char(6)	N	The Processing Code sent in the original transaction (field 3)
TraceNumber	Char(6)	N	Systems Trace Audit Number from original transaction
ReferenceNo	Char(12)	N	Retrieval Reference Number (RRN) assigned by host (field 37)
PSI	Char	Y	Payment Services 2000 Indicator
PSTransID	Char(15)	Y	Payment Services 2000 Transaction Identifier (Generated by VISA)
PSValCode	Char(4)	Y	Payment Services 2000 Validation Code (Generated by VISA)
PSVISAResponseCode	Char(2)	Y	Payment Services 2000 Response code (from VISA)
PSPOSEEntryMode	Char(2)	Y	Payment Services 2000 POS Entry mode (sent to VISA)
AccountSource	Integer	Y	AccountSource defines how the credit card was originally entered when the transaction took place <sup>2</sup>

#### Indexes

Name	Kind	Columns	Purpose
PK_HypercomBatchEntryID	P	HypercomBatchEntryID	Primary key.
IX_HcomBatchEntries_Status_Node_TransDateTime	IX	Status, Node, TransDateTime	Allows efficient identification of pending or most recently posted transactions for a node

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	HYPERCOM_PENDING_STATUS	The payment is still waiting to be settled.
2	HYPERCOM_POSTED_STATUS	The payment has been settled.
3	HYPERCOM_ERROR_STATUS	There was an error attempting to settle the payment.
4	HYPERCOM_VOIDED_STATUS	The payment was voided. This value will not be used for new voided payments, only existing ones already processed.

<sup>2</sup> AccountSource Values

Value	Gateway Constant Name	Description
0	ENDORSEMENT_BLANK	None.
1	ENDORSEMENT_SWIPED	Credit card was swiped.
2	ENDORSEMENT_MANUAL	Credit card entered manually.
3	ENDORSEMENT_SCANNED	Credit card scanned.

### 3.63 HypercomBatchResults

The **HypercomBatchResults** table is used only when authorizing credit card transactions using the Hypercom credit card protocol. It serves as a log of the results of each settlement attempt. The intended use is to facilitate identification of batch settlement problems from SQL, without needing to consult an error or log file. Note that there may be no entry for a batch settlement attempt in the event that a batch attempt failed due to a lack of an SQL connection. Also note that there may be multiple entries for a batch settlement attempt, in the event that one or more settlement attempts failed.

#### Columns

Column	Type	Allow Nulls	Description
HypercomBatchResultID	Int	N	Primary key, always unique.
Node	Int	N	Node number of the Galaxy station.
BatchID	Int	N	3-digit batch ID used for the settlement attempt.
BatchDateTime	DateTime	N	Date and time of the settlement attempt.
SalesCount	Int	N	Number of positive credit card charges included in the batch.
SalesAmount	Money	N	Total of positive credit card charges included in the batch.
ReturnCount	Int	N	Number of credit card returns and voids of sales in the batch.
ReturnAmount	Money	N	Total of negative credit card charges included in the batch.
BatchResult	Int	N	GTS-defined code to summarize result of the settlement <sup>1</sup>
ResponseCode	Char(2)	Y	Response code returned from Hypercom: "00" if settlement attempt was in balance, "95" if a batch upload was required, or an error code if the batch failed.
ResponseFailureText	VarChar(20)	Y	Normally blank, describes the ResponseCode for failed batches.

#### Indexes

Name	Kind	Columns	Purpose
PK_HypercomBatchResultID	P	HypercomBatchResultID	Primary key.

#### <sup>1</sup> BatchResult Values

Value	Gateway Constant Name	Description
0	HYPERCOM_BATCH_FAILURE	The settlement attempt failed, and no authorizations in the batch were settled.
1	HYPERCOM_BATCH_SUCCESS_BALANCE	The settlement attempt succeeded in balance, no batch upload was performed.
2	HYPERCOM_BATCH_SUCCESS_UPLOAD	Original settlement attempt was not in balance, so a batch upload was required. All batch entries were successfully uploaded and settled.
3	HYPERCOM_BATCH_SUCCESS_PARTIAL	Original settlement attempt was not in balance, so a batch upload was required. One or more batch entries in the batch could not be settled.

### 3.64 GalaxyMenus

This table is exported from the B-Tree menudb.dat file via DBSync, and contains Galaxy menu header records.

#### Columns

Column	Type	Allow Nulls	Description
GalaxyMenuID	Int	N	Primary key, always unique
MenuID	Char(16)	Y	Name of the menu
MenuTitle	VarChar(64)	Y	Long description of the menu
OperationID	Int	Y	Reference to Operations.OperationID, specifies the operation to record in a usage log entry if multiple operations were active at the time of the scan. Has no effect unless the Admission Control at POS Validation options "Record Active Operation in Usage" and "Use Operation from POS menu" are configured. Even then, Galaxy uses the corresponding value from the local OPERATON.DAT file rather than the value stored here
RouteID	Int	Y	Stores an optional Route ID for all items in the current menu. This is used only for the RFCS system.
TripID	Int	Y	Stores an optional Trip ID for all items in the current menu. This is used only for the RFCS system.
Inactive	Bit	N	1 if menu is Inactive, and not visible in most picklists.
EffectiveDate	datetime	Y	Date the menu is available for use.

#### Indexes

Name	Kind	Columns	Purpose
PKGalaxyMenuID	P	GalaxyMenuID	Primary Key

### 3.65 GalaxyMenuItems

This table is exported from the B-Tree menitmdb.dat file via DBSync, and contains Galaxy menu detail records.

#### Columns

Column	Type	Allow Nulls	Description
GalaxyMenuItemID	Int	N	Primary key, always unique
MenuID	VarChar(30)	Y	Foreign key to GalaxyMenus.MenuID
ButtonNumber	Int	Y	Button number (1..35)
Kind	Char(8)	Y	Defines the kind of button <sup>1</sup>
HotKey	Char(4)	Y	Hotkey for this button (i.e. 'A', 'F12', etc)
Justify	Char(2)	Y	Justification of the name of the button (R = Right, C=Center, L=Left)
PLU	VarChar(30)	Y	If the Kind is Ticket or Item, PLU associated with this button
Picture	VarChar(255)	Y	Path and filename of picture to display on the button
Price	Float	Y	Price of the ticket/item that this button points to
Category	Char(12)	Y	Currently not used
NameFont	VarChar(32)	Y	Name of the font for the button name
NameColor	Int	Y	Color of the font for the button name
NameSize	Int	Y	Size of the font for the button name
NameBold	Char(2)	Y	If the font for the button name is bolded (Y/N)
NameItalic	Char(2)	Y	If the font for the button name is italicized (Y/N)
PriceFont	VarChar(32)	Y	Name of the font for the button price
PriceColor	Int	Y	Color of the font for the button price
PriceSize	Int	Y	Size of the font for the button price
PriceBold	Char(2)	Y	If the font for the button price is bolded (Y/N)
PricelItalic	Char(2)	Y	If the font for the button price is italicized (Y/N)
HotKeyFont	VarChar(32)	Y	Name of the font for the button hot key
HotKeyColor	Int	Y	Color of the font for the button hot key
HotKeySize	Int	Y	Size of the font for the button hot key
HotKeyBold	Char(2)	Y	If the font for the button hot key is bolded (Y/N)
HotKeyItalic	Char(2)	Y	If the font for the button hot key is italicized (Y/N)
IconFilename	VarChar(255)	Y	Path and filename of icon to display on the button
IconJustify	Char(2)	Y	Justification of the icon (R = Right, C=Center, L=Left)
BackGroundColor	Int	Y	Color to use for background of button
Description	VarChar(30)	Y	Description of button (ticket/item description, function name, etc)
FunctionParameters	VarChar(60)	Y	Any parameters needed for an online or Galaxy function
TransportationDefinitionID	Int	Y	Foreign key to TransportationDefinitions.TransportationDefinitionID
LabelText	VarChar(100)	Y	Alternate text to display on a button
EffectiveDate	datetime	Y	Date the menu if available for use

#### Indexes

Name	Kind	Columns	Purpose
PKGalaxyMenuItemID	P	GalaxyMenuItemID	Primary Key
IXGalaxyMenuItemsPLU		PLU	Used by query to retrieve menu items by PLU

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
Item	btnKindItem	Button is a retail item
Menu	btnKindMenu	Button is a menu
Ticket	btnKindTicket	Button is a ticket
Function	btnKindFunction	Button is a Galaxy function
Onl Func	btnKindOnlFunc	Button is an online function
Display	btnKindDisplay	Button is for display purposes only
Trans	btnKindTransportation	Button is a transportation ticket

### 3.66 GeneratedStatements

Path to statements generated for this contract.

#### Columns

Column	Type	Allow Nulls	Description
GeneratedStatementID	Integer	N	Link to PaymentContract table
StatementType	Integer	Y	Type of Statement: Decline, Renewal, etc. <sup>1</sup>
StatementText	NVarChar(Max)	Y	The text of the statement.
StatementDate	DateTime	Y	Date statement was generated.
PaymentContractID	Integer	Y	Link to Payment Contracts table.

#### Indexes

Name	Kind	Columns	Purpose
PKGeneratedStatementsGenStmtID	P	GeneratedStatementID	Primary Key.

<sup>1</sup> StatementType Values

Value	Gateway Constant Name	Description
0	CONTRACT_STATEMENT_TEMPLATE_TYPE	Statement template
1	CONTRACT_WELCOME_TEMPLATE_TYPE	Welcome template
2	CONTRACT_DECLINE_TEMPLATE_TYPE	Decline template
3	CONTRACT_CONVERT_PLAN_TEMPLATE_TYPE	Convert plan template
4	CONTRACT_CHANGE_CC_TEMPLATE_TYPE	Change credit card template
5	CONTRACT_CC_EXPIRE_TEMPLATE_TYPE	Credit card expired template
6	CONTRACT_RENEWAL_TEMPLATE_TYPE	Renewal template
7	CONTRACT_SUSPEND_TEMPLATE_TYPE	Suspend template

### 3.67 GenericCalendars

The **GenericCalendars** table is used to create calendars that can be associated with various features throughout the system. The rules that indicate when the calendar is valid are stored in the **GenericCalendarDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
GenericCalendarID	Int	N	Primary key, always unique. System generated.
Description	Char(100)	N	Description of the calendar
Inactive	Bit	N	True if the calendar is Inactive, and not visible on most picklists.
GenericCalendarGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGenericCalendarID	P	GenericCalendarID	Primary key.

### 3.68 GenericCalendarDetails

The **GenericCalendarDetails** table contains the rules to specify when a calendar (in the **GenericCalendars** table) is valid.

Note: Calendars currently only support single date inclusive rules. Additional RuleTypes and date ranges, and time frames may be supported in the future.

#### Columns

Column	Type	Allow Nulls	Description
GenericCalendarDetailID	Int	N	Primary key, always unique.
GenericCalendarID	Int	N	Foreign key to GenericCalendars.GenericCalendarID, specifying the calendar that this detail applies to.
StartDate	DateTime	N	The starting date.
EndDate	DateTime	N	The ending date.
StartTime	DateTime	Y	The starting time of the day.
EndTime	DateTime	Y	The ending time of the day.
RuleType	Int	N	Indicates how the dates will be applied to the calendar <sup>1</sup> .
GenericCalendarDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGenericCalendarDetailID	P	GenericCalendarDetailID	Primary key.
IXGenericCalendarDetailCalendarDate		GenericCalendarID,StartDate,EndDate	Used to query the details for a specific calendar

<sup>1</sup> RuleType Values

Value	Gateway Constant Name	Description
0	INCLUDE_DATES	Indicates that dates should be included in the calendar.
1	EXCLUDE_DATES	Indicates that dates should be excluded from the calendar.

### 3.69 GroupSalesLimitHolds

Contains temporary hold data for Group Sales Limits. The hold in this table is already committed e.g. capacity is already reduced before creating this hold. The purpose of this table is to allow applications like Web store to put the hold capacity back to the pool if a guest abandons the session after adding tickets to their cart.

#### Columns

Column	Type	Allow Nulls	Description
GroupSalesLimitHoldID	Int	N	Primary key, always unique
GroupIDType	Int	N	Identifies the contents of the GroupID column. <sup>1</sup>
GroupID	Int	N	A numeric code uniquely identifying the purpose of this record
GroupHoldQty	Int	N	Contains the number Groups on-hold for this hold
GuestHoldQty	Int	N	Contains the number Guests on-hold for this hold
CustCategoryID	Int	Y	FK reference to CustCategories.CustCategoryID
DateTimeHeld	DateTime	N	Date and time of creation of the hold record
GroupVisitDate	DateTime	N	Group visit date for an order/customer
NodeNo	Int	Y	Node number of the machine that added the hold record

#### Indexes

Name	Kind	Columns	Purpose
PKGroupSalesLimitHoldID	P	GroupSalesLimitHoldID	Primary Key.
IXGroupSalesLimitHoldDateGrpID		GroupVisitDate, CustCategoryID, GroupIDType, GroupID	Used by query to retrieve hold records for the given date and type of the group
IXGroupSalesLimitHoldGrpTypeID		GroupIDType, GroupID	Used by query to retrieve hold records by type of the group

<sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	GroupIDType Meaning
0	gsltGALAXY	Hold record is owned by eGalaxy server and GroupID field contains eGalaxySessions.eGalaxySessionID
1	gsltORDERENTRY	Hold record is owned by Order Entry and GroupID field contains Orders.OrderID

### 3.70 GxItemGroups

The GxItemGroups (Gx, meaning Galaxy) table contains the item group definitions that are defined by Galaxy in the local Btree file, ITEMGRP.DAT. These records should not be confused with Order Entry item groups, which are different and maintain strictly in a DBMS.

#### Columns

Column	Type	Allow Nulls	Description
GxItemGroupID	Int	N	Primary key, always unique. System generated.
ItemGroupID	Int	Y	The user definable item group ID number. This value is used to reference an item group throughout the system.
Name	Varchar(20)	Y	The user definable name of the item group / short description.
Description	Varchar(30)	Y	A user definable long description for an item group
Inactive	Bit	N	True if item group is Inactive, and not visible on most picklists.
GxItemGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGxItemGroupsGxItemGroupID	P	GxItemGroupID	Primary key.

### 3.71 GxItemGroupDetails

The GxItemGroupDetails (Gx, meaning Galaxy) table contains the item group detail definitions that are defined by Galaxy in the local Btree file, ITMGRPDT.DAT. These records should not be confused with Order Entry item groups, which are different and maintain strictly in a DBMS.

Each detail in this table links an item PLU to an item group record.

#### Columns

Column	Type	Allow Nulls	Description
GxItemGroupDetailID	Int	N	Primary key, always unique. System generated.
ItemGroupID	Int	Y	The item group ID number this detail is part of. All details with the same ItemGroupID number are part of the same group.
PLU	Char(20)	Y	Foreign key to Items.PLU.
Sequence	Int	Y	Allows users to determine the presentation order during various Pass Portal operations.
GxItemGroupDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGxItemGroupDetailsID	P	GxItemGroupDetailID	Primary key.
AKGxItemGroupDetailsPLU	A	PLU	Used by Export Service module

### 3.72 GxKeys

The **GxKeys** table holds definitions of encryption schemes for use in the Galaxy system. One and only one entry in this table must be marked as the "current" key.

#### Columns

Column	Type	Allow Nulls	Description
GxKeyUniqueID	Integer	N	Primary key, always unique.
GxKeyID	Integer	N	User-defined ID stored with encrypted data.
KeyName	VarChar(100)	N	Name of this key
EncryptionSchemeID	Integer	N	Algorithm used to encrypt data <sup>1</sup>
PassPhrase	VarChar(200)	N	PassPhrase used to encrypt data
CurrentKey	Bit	N	If true, defines this row as the current key to use for encryption.
SystemKeyID	Integer	N	Value defining the version of the encryption scheme that the system uses to encrypt the passphrase.
ExternalKey	VarChar(235)	Y	Column containing external encryption key to use instead of the passphrase. A GxKey entry can either have this field defined or the PassPhrase field defined, but the GUI does not allow both.
ExternalKeyID	VarChar(100)	Y	Column containing the external name associated to the ExternalKey column.
GxKeyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGxKeyUniqueID	P	GxKeyUniqueID	Primary Key.
IXGxKeysGxKeyID	A	GxKeyID	Unique constraint for GxKeyID

#### <sup>1</sup> EncryptionSchemeID Values

Value	Gateway Constant Name	Description
0	AES_RIJNDAEL_256_ENCRYPTION_SCHEME	AES/Rijndael 256-bit (CFB 8-bit Mode)
1	AES_RIJNDAEL_256_ENCRYPTION_SCHEME_CBC_MODE	AES/Rijndael 256-bit (CBC Mode)

### 3.73 GxPrivateKeys

The GxPrivateKeys table holds the keys that are used to encrypt the pass phrases and external keys in the GxMasterKeys table. This was added so Galaxy users can edit the keys used to encrypt other keys.

#### Columns

Column	Type	Allow Nulls	Description
GxPrivateKeyID	Integer	N	Primary key, always unique.
SystemKeyID	Integer	N	ID identifying a private key version. This value will begin with 4. SystemKeyIDs in the GxMasterKeys table with values less than 4 use an internal hardcoded key for encryption. Values of 4 and above identify a GxPrivateKey that was used for encryption. The largest SystemKey ID in the table is used to determine the current private key to be used in the system for encryption.
KeyName	VarChar(100)	N	Name of this private key.
EncryptionSchemeID	Integer	N	Algorithm used to encrypt data <sup>1</sup>
PassPhrase	VarChar(200)	N	Passphrase used to encrypt data. This string is encrypted with an internal encryption key indicated by the InternalKeyID.
InternalKeyID	Integer	N	The internal encryption key version that was used to encrypt the PassPhrase or ExternalKey strings. There can be multiple GxPrivateKey records for the same SystemKeyID, as long as the InternalKeyID is different. These records have the same passphrase or external key, encrypted with a different internal key.
ExternalKey	VarChar(235)	N	Column containing external encryption key to use instead of the passphrase. Either this field or the PassPhrase field can be defined, but the GUI does not allow both. This string is encrypted with an internal encryption key indicated by the InternalKeyID.
ExternalKeyID	VarChar(100)	N	Column containing the external name associated to the ExternalKey column.
GxPrivateKeyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGxPrivateKeysGxPrivateKeyID	P	GxPrivateKeyID	Primary key - Unique ID in the table.
IXGxPrivateKeysSysKIDIntKID	A	SystemKeyID, InternalKeyID	Unique key to prevent the same system key ID from being added for a given internal key version.

#### <sup>1</sup> EncryptionSchemeID Values

Value	Gateway Constant Name	Description
0	AES_RIJNDAEL_256_ENCRYPTION_SCHEME	AES/Rijndael 256-bit encryption (CFB 8-bit Mode)
1	AES_RIJNDAEL_256_ENCRYPTION_SCHEME_CBC_MODE	AES/Rijndael 256-bit encryption (CBC Mode)

### 3.74 GxMasterKeys

The GxMasterKeys table is a replacement for the GxKeys table. It contains definitions of encryption schemes for use in the Galaxy system. One and only one entry in this table must be marked as the "current" key for each SystemKeyID. This table can contain multiple versions of the same key, but with a different encryption scheme used to encrypt the passphrase and the external key. These will have different SystemKeyID values.

#### Columns

Column	Type	Allow Nulls	Description
GxMasterKeyID	Integer	N	Primary key, always unique.
GxKeyID	Integer	N	User-defined ID stored with encrypted data.
KeyName	VarChar(100)	N	Name of this key
EncryptionSchemeID	Integer	N	Algorithm used to encrypt data <sup>1</sup>
PassPhrase	VarChar(200)	N	PassPhrase used to encrypt data
CurrentKey	Bit	N	If true, defines this row as the current key to use for encryption.
SystemKeyID	Integer	N	Value defining the version of the encryption scheme that the system uses to encrypt the passphrase.
ExternalKey	VarChar(235)	N	Column containing external encryption key to use instead of the passphrase. A GxKey entry can either have this field defined or the PassPhrase field defined, but the GUI does not allow both.
ExternalKeyID	VarChar(100)	Y	Column containing the external name associated to the ExternalKey column.
GxMasterKeyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKGxMasterKeysGxMasterKeyID	P	GxMasterKeyID	Primary Key.
IXGxMasterKeysGxKeyIDSysKeyID	A	GxKeyID, SystemKeyID	Unique constraint for GxKeyID and SystemKeyID

<sup>1</sup> EncryptionSchemeID Values

Value	Gateway Constant Name	Description
0	AES_RIJNDAEL_256_ENCRYPTION_SCHEME	AES/Rijndael 256-bit (CFB 8-bit Mode)
1	AES_RIJNDAEL_256_ENCRYPTION_SCHEME_CBC_MODE	AES/Rijndael 256-bit (CBC Mode)

### 3.75 IPAddresses

The **IPAddresses** table holds the valid IP addresses per IP address group.

#### Columns

Column	Type	Allow Nulls	Description
IPAddressID	Integer	No	Primary key, always unique.
IPAddressGroupID	Int	No	Foreign Key into IPAddressGroups table
FromIPAddress	Varchar(15)	No	IP Address or starting IP Address if a range
ToIPAddress	Varchar(15)	No	Ending IP Address in a range.
IPAddressGroupID	Int	No	Foreign Key into IPAddressGroups table
IPAddressGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKIPAddressID	P	IPAddressID	Primary Key.

### 3.76 IPAddressGroups

The **IPAddresses** table contains definitions of IP Address Groups.

#### Columns

Column	Type	Allow Nulls	Description
IPAddressGroupID	Integer	No	Primary key, always unique.
GroupName	Varchar(50)	N	Group Name
Group Description	Varchar(100)	N	Group Description
IPAddressGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKIPAddressGroupID	P	IPAddressGroupID	Primary Key.

### 3.77 ItemGroupReportHdrs

The Item Group Report Definition allows the user to define different report definitions for Sales by Item Group report. The **ItemGroupsReportHdrs** table contains the Item Group Report definition name.

#### Columns

Column	Type	Allow Nulls	Description
ItemGroupReportHdrsID	Int	N	Primary Key, always unique. System generated.
ItemGroupdReportID	Int	N	User definable ID number
Name	VarChar(20)	Y	The name of the Item Group Report definition.

#### Indexes

Name	Kind	Columns	Purpose
PKItemGroupReportHdrID	P	ItemGroupReportHdrsID	Primary key.

### 3.78 ItemGroupReportDtls

The **ItemGroupReportDtls** table contains the details of the Item Group Report Definition such as the caption to use for the report total, the items to include in the report, the name of the report, etc.

#### Columns

Column	Type	Allow Nulls	Description
ItemGroupReportDtlsID	Int	N	Primary key, always unique. System generated.
ItemGroupdReportID	Int	N	Foreign key to ItemGroupReportHdr.ItemGroupdReportID.
ItemGroupdID	Int	N	Foreign key to ItemGroups.ItemGroupdID.
Name	VarChar(20)	Y	The individual Item Group report name.
Abbr	Char(12)	Y	The abbreviation of the report name.
TotLabel	Char(40)	Y	The label used for the total field on the report.
Action	Int	Y	The action to be perform on the item group. <sup>1</sup>
Sequence	Int	Y	The sequence which this Item Group report should print or displayed.

#### Indexes

Name	Kind	Columns	Purpose
PKItemGroupsReportDltID	P	ItemGroupReportDtlsID	Primary key.

#### <sup>1</sup> Action Values

Values	Const Name	Descriptions
1	raAddition	Add to the Grand total for the sales, returns, and amounts.
2	raSubtraction	Subtract from the Grand total for the sales, returns, and amounts.

### 3.79 ItemImages

The ItemImages table contains the images associated with a given item.

#### Columns

Column	Type	Allow Nulls	Description
ItemImageID	Int	N	Primary Key, always unique.
PLU	Char(20)	N	The item to associate this image to.
Origin <sup>1</sup>	Int	N	The format of the ImageData column. <sup>1</sup>
PixelDepth	Int	N	Number of colors ImageData is stored in.
ImageData	Image	N	Bytes on an image
ItemImageGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemImagesItemImageID	P	ItemImageID	Primary key.

#### <sup>1</sup> Action Values

Values	Const Name	Descriptions
0	ORIGIN_BITMAP	
1	ORIGIN_JPEG	
2	ORIGIN_PCX	

### 3.80 Items

The Items table contains the definitions of tickets and items, where items can either be Retail or Food & Beverage items.

#### Columns

Column	Type	Allow Nulls	Description
ItemID	Int	N	Primary key, always unique
PLU	Char(20)	N	This is the PLU for the ticket or the item. This can either be in the form TICKETPPPLLFF for tickets associated with a product, or user-defined for tickets not associated with a product. Also, if it is in the form '\$DISBnnnnnnnnnyy', this is a disbursement information.
DisbursementType	Char(8)	Y	The disbursement ID if the PLU has a disbursement attached to it.
Descr	VarChar(60)	Y	The description of the ticket or the item.
UPC	Char(20)	Y	A number that identify an individual consumer product.
ProductNo	Int	Y	The product number if the PLU is configured to sell from the AX1180 mode.
LevelNo	Int	Y	The level number if the PLU is configured to sell from the AX1180 mode.
FkeyNo	Int	Y	The function key number if the PLU is configured to sell from the AX1180 mode.
AccountIDNo	VarChar(30)	Y	User defined account no external to the system
Company	Int	N	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
Category	Int	N	The transaction account category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SubCat	Int	N	The transaction account sub-category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
Cost	Money	Y	Cost (to retailer)
Price	Money	N	Price (to customer)
Basis	Char(1)	Y	Determines the basis of the Price field: actual, rate, or points. <sup>1</sup>
ValueKind	Int	Y	Currency or Points for the Value field. <sup>2</sup>
Kind	Int	N	This determines whether the PLU is a pass, ticket, item, etc. <sup>3</sup>
StockType	Int	Y	The ticket stock to use for this ticket. Foreign key reference to Stocks.StockID
Printer	Int	Y	The printer number this ticket will print to.
TktSet	Int	Y	The ticket set for this ticket.
DiscID	Int	Y	The ID number of the discount (defined in the Discount table) applied to this ticket, if any.
DiscAmt	Money	Y	The number of dollars, in <i>base currency</i> , discounted from the ticket.
AccessCode	Int	N	The Access Code for the ticket or Pass. Foreign key reference to AccessCodes.AccessCode.
PassKind	Int	Y	The ID of the PassKinds table if the PLU is a pass PLU. Foreign key reference to PassKinds.ID.
PriceEdit	Bit	Y	If TRUE, the ticket seller is allowed to edit the price.
TaxFlag	Int	Y	Bit mask for taxes. The bit is set if taxable for tax 1-8
PictureFlag	Int	Y	Determines whether the picture of the customer must be taken and/or the picture should be saved to the database. <sup>4</sup>
FkeyFlag	Int	Y	Various settings for the ticket. <sup>5</sup>
FOPMask	Char(80)	N	Determines which FOPs are allowed to use as the payment for this ticket. This is populated by populated by 'Y's and 'N's.
Name	VarChar(40)	Y	A name for this PLU.
ModifyKind	Int	Y	This only applies to an item. <sup>6</sup>
ItemFilter	Char(1)	Y	To include when printing when the coupon include @ITEM_FILTER(x) keyword where "x" is the ItemFilter.
EventType	Char(10)	Y	Event type description.
EventID	Int	Y	The EventID associated with the ticket. Foreign key reference to RmEvents.EventID.
SuppressSerial	Bit	Y	If TRUE, the serial of this ticket will be set to zero.
Section	Char(10)	Y	ResourceID to use for the Event associated with this ticket. Foreign key reference to RMResources.ResourceID.
FkeyFlag2	Int	Y	Holds the values for "Allow Sale", "Allow Return" and other miscellaneous item settings. <sup>7</sup>
ItemValue	Int	Y	Points or Dollars on Debit Pass * 100.
PassFlag	Int	Y	Bit mask for pass flags. <sup>8</sup>
ExchangeID	Int	Y	The exchange method used for the ticket exchange. This only applies when the ticket is exchangeable.
Exchange	Int	Y	Ticket Exchange setting (Yes, No, or Today). <sup>9</sup>
ResourceID	Int	Y	This field is not used.
RentalSerialID	Int	Y	This is a FK link to the RentalSerial table.
Serialize	Bit	N	Serialize the rental.
ReservationRequire	Bit	N	If TRUE, reservation for the rental is required.
RentalPhotoRequire	Bit	N	If TRUE, photo is required for the rental.
DateSpecific	Bit	N	If TRUE, the ticket is a date-specific ticket.
BlockoutGroup	Int	Y	Only applies to the DateSpecific tickets. This ticket cannot be sold on the days/dates as specified in the Blockouts entry to which this ticket is associated.
Currency	Char(1)	Y	One character short-cut-key of the currency, as specified in the "Exchange Rate" screen.
DiscountGroup	Char(10)	Y	The discount group this ticket is associated with.
AllowComp	Bit	N	If TRUE, the ticket seller can sell the zero priced tickets.
ReasonRequire	Bit	N	If TRUE, a reason is required to sell this ticket.
TaxMethods	Char(8)	Y	The tax methods is an 8 character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket. <sup>10</sup>
DebitTypeID	Int	N	Foreign key reference to DebitTypes.DebitTypeID
SuppressEvent	Bit	Y	If set to 1, event does not get stored with the disbursed ticket record
ExternalCallID	Int	Y	This is a foreign key link to ExternalCalls.ExternalCallID to external program this item can call.

MaxGroupAdmits	Int	Y	Maximum number per group. The each group ticket will have valid admits upto this amount. If the bought qty is more than this amount, a new ticket will print for the rest upto this amount, and so on.
Status	Int	Y	This value is used as the initial ticket Status value when the ticket gets inserted. This value corresponds to Status column in tickets table. <sup>11</sup>
CanUpgrade	Bit	Y	Indicates if the item can be upgraded
UpgradeValue	Money	Y	Value to be used when upgrading this ticket
DateRange	Bit	Y	Used for Date Range Ticket. Bring up calendar range picker in POS
UseTaxTable	Bit	Y	Indicates if tax tables are to be used for this item
TaxTableID	Int	Y	Value of the tax table header for this item
TaxTableMethod	Int	Y	Indicates if tax is per item(0), or transactional (1)
DeliveryMethodGroupID	Int	Y	FK reference to DeliveryMethodGroups table
TktReqForPurchaseItemGrpID	Int	Y	FK reference to GxItemGroups table
ShortNotes	Varchar(40)	Y	Brief description about the item, used when this item is displayed on the web store
ReplenishPLU	char(20)	Y	The PLU used when the next auto-replenish is processed.
MaxPrice	Money	Y	The maximum price allowable for editable price items
MinPrice	Money	Y	The minimum price allowable for editable price items
RegisterGift	Bit	Y	Indicates the item is a Gift item
CampaignID	Int	Y	Foreign key to Campaigns.CampaignID
FundID	Int	Y	Foreign key to Funds.FundID
PassExpiration	Int	Y	Denotes method to use to compute the pass expiration <sup>12</sup>
RenewalPLU	Char(20)	Y	PLU to use for renewal
PriceMethod	Int	Y	Price method used for the given ticket <sup>13</sup>
SuppressCollectAppealAndSolicitation	Bit	Y	Stores true or false whether the item suppresses collection of appeals and solicitations
MinRate	Int	Y	The minimum rate that can be entered for rate based tickets
MaxRate	Int	Y	The maximum rate that can be entered for rate based tickets
IsLockerPLU	Bit	Y	Set to 1 to sell this PLU as a locker (FKey Retail Item)
LockerFID	Int	Y	FID of the Gantner Locker system. Data entered in this column is used to initialized the skidata smartcard freespace
LockerSubFID	Int	Y	SubFID of the Gantner Locker system. Data entered in this column is used to initialized the skidata smartcard freespace
FeePLU	Char(20)	Y	PLU of the Fee to apply when item sold
UpgradeCompany	Int	Y	Item upgrade variance Chart of Account Company
UpgradeCategory	Int	Y	Item upgrade variance Chart of Account Category
UpgradeSubCategory	Int	Y	Item upgrade variance Chart of Account Sub Category
PromptForName	Bit	Y	Determines if the user should be prompted to enter a guest name when selling a ticket
ContactRequired	Bit	Y	Indicates whether a contact must be selected for a transaction including the item.
AdultExchangeID	Int	Y	The exchange method used for age-specific tickets/Passes for an exchange or return when ticket holder is an Adult.
ChildExchangeID	Int	Y	The exchange method used for age-specific tickets/Passes for an exchange or return when ticket holder is a Child.
GiftAidType	Int	Y	Type of the GiftAid applicable to the item <sup>14</sup>
GiftAidAmount	Money	Y	The amount of GiftAid for the item. When the GiftAid type is percentage, this is a percentage value.
DonationType	Int	Y	Specifies the donation type <sup>15</sup>
RequiresActivation	Boolean	Y	Allows an item to be activated in the POS and Order Entry without entering activation mode
SuppressNameRequired	Boolean	Y	Whether or not suppress requiring a name when selling a Member Add-On product
PhotoAllowed	Boolean	Y	Whether or not a photo can be taken with the sale of the Member Add-On product
MinimumAge	Integer	Y	Minimum age of member using the Add-On product
MaximumAge	Integer	Y	Maximum age of member using the Add-On product
PackagePLU	Char(20)	N	PLU of the package item to associate with the item. This is used for associating an entitlements package to a pass item.
ActivationMethod	Int	Y	Used to indicate the type of validation used when selling an item that requires activation. <sup>16</sup>
ItemPriceMethod	Int	Y	Defines the pricing method for the item <sup>17</sup>
MemberSplitPLU	NVarChar(20)	Y	PLU of the pass item that will replace the original PLU on a joint membership when that membership is split and only one member remains. Typically, this will be used to convert a joint membership into an individual membership when the membership is split.
AttributeValueGroupID	Int	Y	Links Items to the AttributeValues table
EncodeTime	Int	Y	Controls how RFID information is encoded <sup>18</sup>
UpdateRFIDMaps	Bit	Y	True (1) if the RFIDMaps table should be updated with the serial number of any RFID media that is encoded as a result of printing this PLU, otherwise False (0).
PromptForGuestFields	Bit	Y	
GuestFieldAttributeGroupID	Int	Y	
GuestPhotoFlag	Int	Y	For tickets only; defines how guest photo is collected and managed for this item <sup>20</sup> .
IsDualJoint	Bit	Y	True if item is a dual joint membership type.
Inactive	Bit	N	True if item is Inactive, and not visible in most picklists.
AllowOnlineExchange	Bit	Y	Flag to determine if this PLU is one that can be exchanged online, given that other criteria is met.
GuestSaleLimitID	Integer	Y	Links Items to the SaleLimits table for guest sale limits.
PriceProgramGroupID	Integer	Y	Foreign key to PriceProgramGroups.PriceProgramGroupID. Indicates the item is associated with a price program group. Zero indicates no price program group is associated.
InvoiceMethod	Int	Y	For tickets only; determines when tickets generated with this item can be invoiced <sup>19</sup>
OverrideGiftAidAccount	Bit	Y	Determines if the Gift Aid donation will use the GiftAidCompany, GiftAidCategory, and GiftAidSubCat from the item instead of the one in general configuration
GiftAidCompany	Int	Y	The account company to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
GiftAidCategory	Int	Y	The account category to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
GiftAidSubCat	Int	Y	The account sub-category to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
KeepAccount	Bit	Y	Maintain original sale revenue in this items account.

OverrideGiftAidAccount	Bit	Y	Determines if the Gift Aid donation will use the GiftAidCompany, GiftAidCategory, and GiftAidSubCat from the item instead of the one in general configuration.
GiftAidCompany	Int	Y	The account company to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
GiftAidCategory	Int	Y	The account category to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
GiftAidSubCat	Int	Y	The account sub-category to use for the Gift Aid donation. Only used if OverrideGiftAidAccount is True(1).
DualMembershipPLU	NVarChar(20)	Y	The PLU of a dual membership product that must be sold along with this product in a transaction.
EnableSaleStartDate	Bit	Y	Determines if the sale of this product is limited by the sale start date.
EnableSaleEndDate	Bit	Y	Determines if the sale of this product is limited by the sale end date.
SaleFrom	DateTime	Y	The start date for sale of this product.
SaleThru	DateTime	Y	The end date for sale of this product.
PriceSchedulePricingPluginID	Int	Y	Foreign key to Plugins.PluginID. Indicates the plugin to use for price schedule pricing.
AllowReturnUntilLastUsage	Bit	Y	Determines if returns on this item should be restricted to only allow returns until day of last usage.
AllowReturnUntilTicketExpires	Bit	Y	Determines if returns on this item should be restricted to only allow returns until the ticket is expired.
SaleGenericCalendarID	Int	Y	Foreign key to GenericCalendars.GenericCalendarID. Indicates the calendar to use for sale of this product.
PreventReturnOfUsedTickets	Bit	Y	Determines if returns on this item should be restricted for tickets with usage.
PreventEntitlementAddOns	Bit	Y	Determines if an entitlement add-on or deferred entitlement add-on can be added to the item.
SkipEventSelectionForInactiveItems	Bit	Y	Indicates if event selection will be skipped when selling tickets with an inactive status in Galaxy.
SkipDemographicPromptsForInactiveItems	Bit	Y	Indicates if prompts for guest demographic information will be skipped when selling tickets with an inactive status in Galaxy.
EventModificationFromDays	Int	Y	The number of days before a visit when changes can be made to an existing reservation. Used only if OverrideCentralExchangeRules is set to 1.
EventModificationThruDays	Int	Y	The number of days before a visit when changes to an existing reservation are no longer allowed. Used only if OverrideCentralExchangeRules is set to 1
MaxEventModifications	Int	Y	Determines how many changes can be made to an existing reservation. Used only if OverrideCentralExchangeRules is set to 1.
OverrideCentralExchangeRules	Bit	Y	Determines if the EventModificationFromDays, EventModificationThruDays and MaxEventModifications values are used instead of the values from General Configuration in Central Data Maintenance.
ActivateByCalendarID	Int	Y	ID of access calendar used to determine the date at which an inactive item must be activated by.
OverrideEndOfLifeDateRules	Bit	Y	Indicates the end of life date settings on the item will be used.
EndableEndOfLifeDateLock	Bit	Y	Indicates if the item will become locked when the end of life date window passes.
EndOfLifeDateWindowDays	Int	Y	The number of days after the end of life date when the item becomes locked.
ItemGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**Indexes**

Name	Kind	Columns	Purpose
PKItemsItemID	P	ItemID	Primary Key.
AKItemsPLU	A	PLU	Alternate Key
IXItemsPackagePLU	IX	PackagePLU	Index to enhance lookup of pass items with associated packages.
IXItemsLastUpdate	IX	LastUpdate	Only created when Smart Upsell is enabled to help with dynamic configuration loading.
IXItemsInactive	IX	Inactive	Index to enhance lookup filtering by active items.

**1 Basis Values**

Value	Description
V	The price is the actual item value.
R	The price is a rate.
Blank Space	The price is a point value

**2 ValueKind Values**

Value	Description
0	Currency
1	Points

**3 Kind Values**

Value	Const Name	Description
1	PLAIN_FKEY	Regular/Ticket
2	PASS_FKEY	Pass
3	PAID_IN_FKEY	Paid In
4	PAID_OUT_FKEY	Paid Out
5	GROUP_FKEY	Group
6	REISSUE_PASS_FKEY	Reissue Pass
7	RENEW_PASS_FKEY	Renew Pass
8	ITEM_FKEY	Item
9	RESERVED_FKEY	Reserved
10	DEBIT_FKEY	Debit Card
11	RECHARGE_FKEY	Recharge
12	REISSUE_DEBIT_FKEY	Debit Reissue
13	UPGRADE_PASS_FKEY	Upgrade Pass
14	EXCHANGE_FKEY	Exchange
15	PACKAGE_FKEY	Package
16	SEAT_ASSIGNED_FKEY	Assigned Seat

17	FEE_FKEY	Fee
18	DONATION_FKEY	Donation
19	EXPRESS_CHARGE_FKEY	Express charge
20	MEMBER_ADD_ON_FKEY	Member add-on
21	REISSUE_CARD_FKEY	Member card reissue
22	FOOD_BEVERAGE_FKEY	Food and Beverage item

**4 PictureFlag Values**

Value	Const Name	Bit Position	Description
128	PICTURE_STORAGE (1)	1000 0000	When this option is set, the picture is stored in the database
1	PICTURE_REQUIRED (8)	0000 0001	When this option is set, the ticket or pass requires a picture to be taken.

**Note:** The values indicated above are the basis for the value in the PictureFlag field. The actual PictureFlag value in the Items table entry might not be same as the ones indicated above. In this case, the value is a combination of the above values. As an example, if the value of the PictureFlag is 129 (128 + 1), the customer's picture must be taken and the picture will be saved to the database.

**5 FkeyFlag Values**

Value	Const Name	Bit Position	Description
128	FKEY_INCL_TAX (1)	1000 0000	When this option is set, the price includes tax
64	FKEY_CUST_ORDER (2)	0100 0000	When this option is set, the customer order is loaded automatically
32	FKEY_CHECK_PASS (3)	0010 0000	When this option is set, Galaxy prompts for pass, for single use per day
16	FKEY_CUSTOMER (4)	0001 0000	When this option is set, the transaction for this ticket requires customer
8	FKEY_DISC_REQ (5)	0000 1000	When this option is set, a discount is required before end of the transaction
4	FKEY_MULTI_PASS (6)	0000 0100	When this option is set, Galaxy Prompts for pass, for unlimited use per day
2	FKEY_PRE_PRINTED (7)	0000 0010	When this option is set, this ticket is a pre-printed ticket
1	FKEY_LOAD_CUST (8)	0000 0001	Mutually exclusive w/ FKEY_CUST_ORDER

**Note:** The values indicated above are the basis for the value in the FkeyFlag field. The actual FkeyFlag value in the Items table entry might not be same as the ones indicated above. In this case, the value is a combination of the above values. As an example, if the value of the FkeyFlag is 66 (64 + 2), the customer order will be loaded automatically and the ticket would be printed on a pre-printed ticket.

**6 ModifyKind Values**

Value	Const Name	Description
0	DISALLOW_MODIFIER	Item may not be modified / Regular item with no modifiers
1	ALLOW_MODIFIER	Item may or may not accept modifier
2	ENFORCE_MODIFIER	Item must have a modifier (not implemented)
3	IS_MODIFIER	This item is a modifier
4	IS_MODIFIED	This item is the result of a modified item

**7 FkeyFlag2 Values**

Value	Const Name	Bit Position	Description
128	SELL_WITH_APPROVAL (1)	1000 0000	The supervisor's approval is required to sell this ticket.
64	NEVER_SELL (2)	0100 0000	This ticket cannot be sold.
N/A	N/A	00xx xxxx	Always allow the sale of this ticket if the first two bit positions are set to zero.
32	RETURN_WITH_APPROVAL (3)	0010 0000	The supervisor's approval is required to return this ticket.
16	NEVER_RETURN (4)	0001 0000	This ticket cannot be returned.
N/A	N/A	xx00 xxxx	Always allow the return of this ticket if the bit positions 3 and 4 are set to zero.
8	IS_RESERVATION (5)	0000 1000	This ticket is a Resource Reservation.
4	N/A (6)	0000 0100	NOT USED
2	IS_CAPACITY (7)	0000 0010	This ticket is Capacity Managed.
1	BULK_EXCHANGE (8)	0000 0001	Used internally to mark an item for bulk ticket exchange.

**Note:** The values indicated above are the basis for the value in the FkeyFlag2 field. The actual FkeyFlag2 value in the Items table entry might not be same as the ones indicated above. In this case, the value is a combination of the above values. As an example, if the value of the FkeyFlag2 is 144 (128 + 16), this ticket must be approved by a supervisor to complete the transaction and this ticket cannot be returned.

**8 PassFlag Values**

Value	Const Name	Bit Position	Description
128	SAME_PASS_KIND_FLAG (1)	1000 0000	Use the same pass kind as the original pass on reissue
64	RESET_EXPIRATION_FLAG (2)	0100 0000	Recompute the pass expiration date on reissue
32	FULL_JOURNALIZATION_FLAG (3)	0010 0000	Use three journal ticket records on reissue
16	PASS_RETURN_FLAG (4)	0001 0000	Use a picklist to select a pass on returns
8	USE_OLD_PASS_ID_FLAG (5)	0000 1000	Always use old pass ID on reissue, regardless of passkind settings

**Note:** The values indicated above are the basis for the value in the PassFlag field. The actual PassFlag value in the Items table entry might not be same as the ones indicated above. In this case, the PassFlag value is a combination of the above values. As an example, if the PassFlag value is 160 (128 + 32) the re-issued pass' pass kind will be same as the original pass and there will be full journalization of the pass information.

**9 Exchange Values**

Value	Const Name	Description
0	EXCHANGE_YES	Ticket exchange is allowed

1	EXCHANGE_NO	Ticket exchange is not allowed
2	EXCHANGE_TODAY	Ticket exchange is only allowed for the same day

**10 TaxMethod values, per character**

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis

**11 Status Values**

Value	Gateway Constant Name	Description
0	TKT_VALID	The ticket is valid or active
3	TKT_INACTIVE	The ticket is inactive

**12 PassExpiration Values**

Value	Gateway Constant Name	Description
0	DONT_RESET_EXPIRATION	Do not reset expiration date
1	ALWAYS_RESET_EXPIRATION	Always reset the expiration date
2	RESET_EXPIRED_EXPIRATION	Reset the expiration date for expired passes

**13 PriceMethod Values**

Value	Gateway Constant Name	Description
0	BASE_PRICE_PRICE_METHOD	Price of the ticket = Base price configured on the ticket. Price schedule is NOT used
1	DATE SOLD PRICE METHOD	Price of a ticket is determined by the date ticket is sold, using the Price schedule
2	USE TIME PRICE METHOD	Price of a ticket is determined by the date ticket is used, using the Price schedule
3	VISIT DATE PRICE METHOD	Price of a ticket is determined by the visit date on the ticket, using the Price schedule

**14 GiftAidType Values**

Value	Gateway Constant Name	Description
0	gaNone	No GiftAid allowed for the item.
1	gaFixedAmount	Fixed amount of Gift Aid to add to the item
2	gaFull	Full ticket price can be used as GiftAid
3	gaNonGAFixedAmount	Non-Gift Aid donation is a fixed amount for the ticket price based on the configured percentage
4	gaNonGAFull	Non-Gift Aid donation amount is full ticket price
5	gaPercentage	Calculate Gift Aid amount as a percentage of the item price
6	gaNonGAPercentage	Non-Gift Aid donation amount calculated as a percentage of the item price

**15 DonationType Values**

Value	Gateway Constant Name	Description
0	DONATION_TYPE_NONE	The item has no donation type
1	DONATION_TYPE_ROUND_UP	When this item is added to a transaction, the transaction total is rounded up to the nearest whole dollar; the round-up amount becomes the price of the item and is the amount of the donation. In this case, the price of the item is not editable by the user.
2	DONATION_TYPE_FLAT RATE	When this item is added to a transaction, the price of the item itself is the donation amount. The price of the item may or may not be editable by the user depending on the item configuration.

**16 ActivationMethod Values**

Value	Const Name	Description
0	ITEM_ACTIVATION_REQUIRE_INACTIVE_MEDIA	This value indicates that the media being activated must exist in the Tickets table with a status of inactive. This option is restricted for use only with regular (ticket) items.
1	ITEM_ACTIVATION_REQUIRE_ANY_MEDIA	This value indicates that any media can be used to activate the given item, even if it does not exist in the Tickets or Passes table.
2	ITEM_ACTIVATION_REQUIRE_VALIDATE_BY_MEDIA_DEF	This value indicates that any media can be used to activate the given item. Media will be validated against the item's media definition.

**17 ItemPriceMethod Values**

Value	Const Name	Description
0	ITEM_PRICE_METHOD_STATIC	Item uses a single price that never changes.
1	ITEM_PRICE_METHOD_EFFECTIVE_DATE	Item has associated entries in the ItemPrices table that define the item's price by effective date.

**18 EncodeTime Values**

Value	Const Name	Description
0	etDefault	RFID information should be encoded in the traditional manner (varies depending on what is being encoded)
1	etEncodeWhilePrinting	RFID information should be encoded while printing a coupon. This requires a printer capable of encoding while printing.
2	etEncodeAfterPrinting	RFID information should be encoded after printing a coupon. This is typically used when printing with one device and encoding with another device.

**19 InvoiceMethod Values**

Value	Const Name	Description
0	INVOICE_METHOD_ON_SALE	Ticket can be invoiced when sold
1	INVOICE_METHOD_ON_USE	Ticket can be invoiced when used

**20 GuestPhotoFlag Values**

Value	Const Name	Bit Position	Description
128	GP_GUEST_PHOTO_REQUIRED (1)	1000 0000	When this option is set, the item requires a guest photo to be taken.
64	GP_PROMPT_AT_POS (2)	0100 0000	When this option is set, the system will prompt for a guest photo to be taken when the ticket is printed. Otherwise, the system will prompt for a guest photo to be taken when the ticket is used.
32	GP_COLLECT_GUEST_NAME_AFTER_PHOTO_DELETED (3)	0010 0000	When this option is set, the system will prompt for guest name when the guest photo is deleted.
16	GP_COLLECT_GUEST_FIELDS_AFTER_PHOTO_DELETED (4)	0001 0000	When this option is set, the system will prompt for guest field attributes when the guest photo is deleted. Uses GuestFieldAttributeGroupId for guest field attributes.

### 3.81 ItemConnections

The ItemConnections table is used to externally link an item to another entity. What exactly the item is linked to is determined by the ConnectionKind column which defines the type of connection, and the ConnectionID which represents a foreign key to the UniqueID of the associated table.

#### Columns

Column	Type	Allow Nulls	Description
ItemConnectionID	Int	N	Primary key, always unique
PLU	Char(20)	N	The PLU of the item
ConnectionKind <sup>1</sup>	Int	N	Defines the kind of connection (see values below)
ConnectionID	Int	N	Represents a foreign key to the UniqueID of the associated table as defined by the ConnectionKind field.
ItemConnectionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemConnectionID	P	ItemConnectionID	Primary Key.

#### <sup>1</sup> ConnectionKind Values

Value	Gateway Constant Name	Description
1	PRINT_AT_HOME_TEMPLATE	ItemConnection points to an entry in the eGalaxyTemplates table where ConnectionID = eGalaxyTemplates.eGalaxyTemplateID.
2	PAYMENT_PLAN	ItemConnection points to an entry in the PaymentPlans table where ConnectionID = PaymentPlans.PaymentPlanID.
3	ACTIVITY_TEMPLATE	ItemConnection points to an entry in the Activities table where ConnectionID = Activities.ActivityID.
4	ITEM_CONNECTION_KIND_ADD_ON_PASS_KIND	ItemConnection points to an entry in the PassKinds table where ConnectionID = PassKinds.ID.
5	APPLE_WALLET_TEMPLATE	ItemConnection points to an entry in the eGalaxyTemplates table where ConnectionID = eGalaxyTemplates.eGalaxyTemplateID.

### 3.82 ItemPrices

The ItemPrices table defines a set of one to many effective date prices to a specific PLU. A PLU cannot have more than one entry per EffectiveDate. Entries in this table define an effective price for a PLU on a specific date.

#### Columns

Column	Type	Allow Nulls	Description
ItemPriceID	Int	N	Primary key, always unique
EffectiveDate	DateTime	N	Date this price takes effect
PLU	NVarChar (20)	N	The PLU of the item associated to this entry (FK to Items.PLU)
Price	Money	N	The effective price for the item.
Cost	Money	Y	The cost of this item.
MinPrice	Money	Y	The minimum editable price for this item.
MaxPrice	Money	Y	The maximum editable price for this item.

#### Indexes

Name	Kind	Columns	Purpose
PKItemPricesItemPriceID	P	ItemPriceID	Primary Key
IXItemPricesPLUEffectiveDate	IX	PLU, EffectiveDate	Unique constraint to avoid duplicate entries for the same PLU on the same effective date.

### 3.83 ItemModifierGroupDetails

The ItemModifierGroupDetails table defines an association of modifier groups to an item. These groups are prompted for when the item is sold in the Galaxy Point of Sale. This table also defines rules for selection of the modifiers.

#### Columns

Column	Type	Allow Nulls	Description
ItemModifierGroupDetailID	Int	N	Primary key, always unique
ItemPLU	NVarChar(20)	N	Foreign key to Items.PLU. The item that the modifier group is associated with.
ModifierGroupID	Int	N	Foreign key to ModifierGroups.ModifierGroupID. Indicates the modifier group that is associated with the item.
SelectMax	Int	Y	Indicates the maximum number of modifiers that can be selected from the group. A value of 0 indicates that there is no maximum. Overrides the value on the modifier group. This field is only used if OverrideGroupMinMax is set to true.
SelectMin	Int	Y	Indicates the minimum number of modifiers that can be selected from the group. Overrides the value on the modifier group. This field is only used if OverrideGroupMinMax is set to true.
Sequence	Int	Y	Indicates the sequence in which the groups are displayed to the user for modifier selection. The first group shown has a sequence of 0.
OverrideGroupMinMax	Bit	Y	Indicates if the minimum and maximum selections defined in the SelectMin and SelectMax fields in this table will override the value defined on the modifier group. A value of 1 indicates that the override values will be used.

#### Indexes

Name	Kind	Columns	Purpose
PKItemModifierGroupDetailID	P	ItemModifierGroupDetailID	Primary key
IXItemModifierGroupDetailsItemPLU	IX	ItemPLU	Used to load the modifier groups associated with a specified item.
IXItemModifierGroupDetailsModifierGroupID	IX	ModifierGroupID	Used to load the items that have the specified modifier group associated.

### 3.84 ItemStatements

This table contains the list of eGalaxyTemplates that will print when an item is purchased.

#### Columns

Column	Type	Allow Nulls	Description
ItemStatementID	Int	N	Primary key, always unique.
PLU	Varchar(20)	N	Foreign key to Items table
Printer	Varchar(256)	N	The name of the windows printer where the statement will print
TemplateID	Int	N	Foreign key to eGalaxyTemplates.eGalaxyTemplateID. This specifies the PDF template to be used to generate the statement to be printed.

#### Indexes

Name	Kind	Columns	Purpose
PK ItemStatementID	P	ItemStatementID	Primary Key.

### 3.85 ItemTranslations

The ItemTranslations table stores translations of the Item's Description field for specific languages. This table can be expanded if any other fields on an item need to be translated in this manner.

#### Columns

Column	Type	Allow Nulls	Description
ItemTranslationID	Int	N	Primary key, always unique
PLU	Char(20)	N	Foreign key to Items table
Language	Char(2)	N	References the ISO 639-1 language code
Description	NVarchar(256)	Y	Contains the Unicode description of the item in the appropriate language
Name	NVarchar(256)	Y	Contains the Unicode name of the item in the appropriate language
ItemTranslationGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemTranslationsItemTranslationID	P	ItemTranslationID	Primary Key
IXItemTranslationsPLULanguage	F	PLU, Language	For lookups by PLU or PLU and language

### 3.86 ItemUpgrades

#### Columns

Column	Type	Allow Nulls	Description
ItemUpgradeID	Int	N	Primary key, always unique
ItemPLU	nvarchar (20)	Y	The PLU of the item to be upgraded
UpgradePLU	nvarchar (20)	Y	The PLU used to upgrade the ItemPLU item.
ItemGroupID	Int	Y	FK to GxItemGroups.ItemGroupID, used to upgrade the ItemPLU item.
Sequence	Int	Y	Allows user to determine presentation order during Pass Portal operations.
UpgradeEffectiveDate	DateTime	Y	Date the upgrade becomes available for sale.
UpgradeExpirationDate	DateTime	Y	Date the upgrade expires and is no longer available.
UpgradeDescription	nvarchar (255)	Y	Description that will be used on the kiosk to describe the upgrade
UpgradeRequiresSourceUsage	Bit	Y	Upgrade is only available if the source product has been used
UpgradeUntilSourceExpired	Bit	Y	Upgrade is available until the source product has expired
UpgradePreventDowngrade	Bit	Y	Prevent downgrades.
SalesChannelID	Int	Y	FK to SalesChannels.SalesChannelID, used to determine where an upgrade is allowed. Zero indicates no Sales Channel is associated.
UpgradePreventDowngradeForUsedTickets	Bit	Y	Prevent downgrades for used tickets.
ScriptTemplateID	Int	Y	Web template ID of the script to be used for this upgrade option
Hide	Bit	N	Tells if this option should be hidden from users when upgrade options are displayed, defaults to FALSE
Emphasis <sup>1</sup>	Int	Y	How this option should be emphasized when upgrade options are displayed, defaults to 0
UpgradeType <sup>2</sup>	Int	N	Whether this is an upgrade replacement or an upgrade add-on, defaults to 0 (upgrade replacement)
PreventFundReturnForDowngrade	Bit	Y	Prevents the return of funds during a ticket downgrade.

#### Indexes

Name	Kind	Columns	Purpose
PKItemUpgradeID	P	ItemUpgradeID	Primary Key.
IXItemUpgradesLastUpdate	IX	LastUpdate	Only created when Smart Upsell is enabled to help with dynamic configuration loading.

#### <sup>1</sup> Emphasis Values

| Value | Gateway Constant Name | Description | |-----|-----|-----| | 0 | Normal | This option is listed as normal | | 1 | Low | This option should be deemphasized | | 2 | High | This option should be emphasized | ### <sup>2</sup> UpgradeType Values {-}

Value | Gateway Constant Name | Description |  
0 | UPGRADE\_TYPE\_REPLACEMENT | This upgrade item replaces the original item when chosen |  
1 | UPGRADE\_TYPE\_ADD\_ON | This upgrade item is added-on to the original as an entitlement add-on |

### 3.87 ItemVariations

This table will be used to select Item Variations that will be available on the Web store.

#### Columns

Column	Type	Allow Nulls	Description
ItemVariationID	Int	N	Primary key, always unique
PLUSource	Char(20)	N	PLU that is the starting point for the variation
PLUVariant	Char(20)	N	PLU to be used as the variation if this variation is selected
Description	VarChar(255)	N	Optional field - additional information (used on Web Store)
DefaultVariation	Bit	N	Determine if this record will be used as the default record for this BasePLU
ItemVariationGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemVariationID	P	ItemVariationID	Primary Key.

### 3.88 Itineraries

This table stores the itinerary including information on the carrier, schedule and departure time. An itinerary is a schedule specific instance of a list of legs.

#### Columns

Column	Type	Allow Nulls	Description
ItineraryID	Int	N	Primary key, always unique
Modified	Boolean	Y	True if itinerary was edited by user
Carrier	Char(4)	Y	First carrier out
Sched	Char(4)	Y	First leg's schedule
Departs	DateTime	Y	Departure time of the first leg
Arrives	DateTime	Y	Arrival time of the last leg
Freq	Char(8)	Y	Days the itinerary operates
ET	Int	Y	Total elapsed time in minutes including layovers
Miles	Int	Y	Total miles
Stops	Int	Y	Total number of stops
Breaks	Int	Y	Number of schedule changes
AvgPTP	Float	Y	Average of PTP's for the via points
FareRouteIndex	Int	Y	
TariffMilesRouteIndex	Int	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKItineraryID	P	ItineraryID	Primary Key.

### 3.89 ItineraryDetails

The legs table defines the origin, destination, carrier and schedule for an individual leg. A trip may contain multiple legs. Legs are a list of city pairs.

#### Columns

Column	Type	Allow Nulls	Description
ItineraryDetailID	Int	N	Primary key, always unique
ItineraryID	Int	N	Foreign Key to Itineraries table
LegNo	Int	N	Order of legs in trip
Origin	Int	N	Origin information for this leg
Destin	Int	N	Destination information for this leg
ValidSched	Boolean	Y	True if carrier/schedule is on database
CompanyNum	Int	Y	Company number for carrier
Carrier	Char(4)	N	Carrier for this leg
Sched	Char(4)	N	Carrier's schedule number
EffDate	DateTime	N	Schedule's effective date
ExpDate	DateTime	Y	Schedule's expiration date
Departs	DateTime	Y	Depart time
Arrives	DateTime	Y	Arrive time
Layover	Int	Y	Layover time in minutes
ET	Int	Y	Elapsed time from departs to arrive for this leg
Stops	Int	Y	Number of stops on schedule for this leg
Seg1	Int	Y	Segment number of the origin on schedule
Seg2	Int	Y	Segment number of the destin on schedule
Miles	Int	Y	Schedule miles from origin to destin
Freq	Char(8)	Y	Days the schedule operates
ThruFreq	Char(8)	Y	Days we actually arrive at the next leg
CouponValue	Int	Y	Coupon value x 100 to use for this leg
CVMiles	Int	Y	Miles used to calculate the coupon value
SeatRecordId	Int	Y	Primary key in TSA journal
SeatDesignator	Char(4)	Y	Seat number from reserved seat mode

#### Indexes

Name	Kind	Columns	Purpose
PKItineraryDetailID	P	ItineraryDetailID	Primary Key.

### 3.90 LeadMessages

This table contains the master copy of messages for both Lead (POS) and Tlead (Turnstiles).

#### Indexes and Constraints

Primary Key: PKLeadMessageID

Indexes:

(None)

Column	Type	Allow Nulls	Description
LeadMessageID	Int	N	Primary key, always unique
Active	Bit	N	Flag indicating whether or not the message is active
Descr	Char(25)	N	Message description
Lines	Char(200)	Y	Message text
Urgent	bit	N	Flag to indicate the message is urgent (POS messages only)
Kind	Int	N	Indicates who uses this message, POS or ACP <sup>1</sup>

#### <sup>1</sup> Kind Values

Value	Description
1	Both POS and Access Control can use this message (this value is for future use)
2	The message is for POS Nodes only
3	The message is for Access Control Points (turnstiles) only

### 3.91 LockerAssignments

#### Columns

Column	Type	Allow Nulls	Description
LockerAssignmentID	Integer	N	Unique Table ID
LockerID	Integer	N	Locker table ID - Foreign key reference to Lockers.LockerID
VisualID	Varchar(40)	N	VisualID used to lookup corresponding RFID in RFIDMaps - Foreign key reference to RFIDMaps.VisualID
Status	Integer	N	Status of the locker (0 = In Use, 1= No longer used by this VisualID)
AssignmentDate	DateTime	N	Locker assignment date. Typically the date when the locker transaction took place

#### Indexes

Name	Kind	Columns	Purpose
PKLockerAssignmentsLAID	P	LockerAssignmentID	Unique Table ID
IXLockerAssignmentsStatusDate		Status, AssignmentDate	Used by query to update status of a locker at the end of the day

### 3.92 LockerLog

#### Columns

Column	Type	Allow Nulls	Description
LockerLogID	Integer	N	Unique Table ID
LockerID	Integer	N	Locker table ID - Foreign key reference to Lockers.LockerID
SerialNumber	Char(20)	N	Chip serial number
VisualID	VarChar(40)	Y	VisualID associated with chip if available
LogType	Integer	Y	Type of log entry <sup>1</sup>
TransactionAmount	Money	Y	Amount of the transaction
Message	VarChar(255)	Y	Text area for error logging/messages

#### Indexes

Name	Kind	Columns	Purpose
PKLockerLogLockerLogID	P	LockerLogID	Unique Table ID

<sup>1</sup> LogType Values

Value	Gateway Constant Name	Description
0	LOCKER_LOG_REJECTED_TRANSACTION	LogType column value for a rejected transaction.
1	LOCKER_LOG_NEGATIVE_BALANCE	LogType column value for a negative balance.
2	LOCKER_LOG_COMM_DISCONNECTED	LogType column value for a comm.. Disconnection.
3	LOCKER_LOG_LOCKER_NOT_FOUND	LogType column value for a locker not found error in Galaxy Locker Server

### 3.93 Lockers

#### Columns

Column	Type	Allow Nulls	Description
LockerID	Integer	N	Unique Table ID
PLU	Char(20)	N	Locker table ID - Foreign key reference to Lockers.LockerID
ExternalID	Char(20)	N	Locker ID used by Gantner locker system to identify this locker
LockerDescription	VarChar(80)	Y	Description of the locker
LockerNumber	Integer	Y	Physical number on the locker
LockerTypeID	Integer	Y	Foreign key reference to LockerTypes.LockerTypeID

#### Indexes

Name	Kind	Columns	Purpose
PKLockersLockerID	P	LockerID	Unique Table ID
IXLockersExternalIDTypeID		ExternalID, LockerTypeID	Used by query to get PLU associated with the given External Locker ID and locker type

### 3.94 LockerTypes

LockerTypes table will be used for storing Locker Types that will be tied to Lockers through the LockerTypeID. The Description field will hold a brief description of the Locker Type and the Abbreviation holds an abbreviation to be associated with the Locker Type. This will allow for men's lockers to be designated as M1, M2 M3 and women's lockers to be designated as W1, W2, W3 etc. Both the description and the abbreviation are optional and nulls are allowed and furthermore Lockers are not required to be tied to a Locker Type.

This table was needed due to the numbering scheme that Wild Wadi and the Gantner locker system utilize. There needed to be an easy way to differentiate between Lockers with the same number, but of different types as the example above demonstrates.

#### Columns

Column	Type	Allow Nulls	Description
LockerTypeID	Integer	N	Unique ID
Description	VarChar(20)	Y	Description of the locker type
Abbreviation	VarChar(10)	Y	Abbreviation for the locker type

#### Indexes

Name	Kind	Columns	Purpose
PKLockerTypesLockerTypeID	P	LockerTypeID	Unique ID

### 3.95 LoyaltyAccountLog

The LoyaltyAccountLog table table contains a summary line for each loyalty transaction in the sql journal.

Column	Type	Allow Nulls	Description
LoyaltyAccountLogID	Integer	N	Primary Key, always unique
AccountNo	Varchar(40)	Y	Loyalty account number
LoyaltyProgramID	Integer	N	LoyaltyProgramID associated with the account
JnlTranID	Integer	N	Unique id for the journal transaction, foreign key to JnlHeaders
TranNo	Integer	N	Transaction number journalized from the POS
NodeNo	Integer	N	The POS node where transaction occurred
TranDate	DateTime	N	Date of the transaction
Points	Float	N	Loyalty points in the transaction
LogType	Integer	N	Type of log entry <sup>1</sup>
PointExpirationDate	DateTime	Y	The Date on which the points earned in the transaction will expire.
PointsUsed	Float	Y	The number of points used via redemption for this log entry. Only valid when the log type is 0 (accrual).
UserID	Integer	Y	User responsible to this log entry, FK to GxUsers.UserID
Reference	VarChar(255)	Y	When an balance adjustment is made, this column will contain the required reason. For a LoyaltyRedemption eGalaxy Server message, this column will contain the order's ExternalID value.

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyAccountLogID	P	LoyaltyAccountLogID	Primary Key

#### <sup>1</sup> LogType Values

Value	Gateway Constant Name	Description
0	LOYALTY_LOG_TYPE_ACCRUAL	Log entry for accrued points
1	LOYALTY_LOG_TYPE_REDEMPTION	Log entry for redeemed points
2	LOYALTY_LOG_TYPE_EXPIRED	Log entry for expired points
3	LOYALTY_LOG_TYPE_BONUS	Log entry for bonus points
4	LOYALTY_LOG_TYPE_ADJUST	Log entry for a balance adjustment

### 3.96 LoyaltyAccounts

The LoyaltyAccounts table contains the individual loyalty accounts that are linked to contacts.

Column	Type	Allow Nulls	Description
LoyaltyAccountID	Int	N	Primary key, always unique
ContactID	Int	N	Contact this loyalty account is associated to, FK to CustContacts.CustContactID
LoyaltyProgramID	Int	N	Loyalty program used to create this account, FK to LoyaltyPrograms.LoyaltyProgramID
AccountNo	VarChar(40)	N	Unique, generated ID associated with this loyalty account. Generated in a similar fashion as a VisualID
JoinDate	DateTime	N	Date/Time the member signed up for the account

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyAccountID	P	LoyaltyAccountID	Primary Key

### 3.97 LoyaltyAccrualItems

The LoyaltyAccrualItems table defines what PLUs can earn points for the loyalty program, and the point value associated to each PLU.

Column	Type	Allow Nulls	Description
LoyaltyAccrualItemID	Int	N	Primary key, always unique
LoyaltyProgramID	Int	N	FK to LoyaltyPrograms.LoyaltyProgramID
PLU	Char(20)	N	Ticket/Item PLU, FK to Items.PLU
AccrualPointValue	Float	N	The number of points earned when purchasing this PLU
AutoEnroll	Bit	N	Flag used to force an enrollment into a loyalty program during a transaction
LoyaltyProgramLevelID	Int	Y	Program Level the item is associated with; 0 indicates no assigned level
LoyaltyAccrualItemGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyAccrualItemID	P	LoyaltyAccrualItemID	Primary Key

### 3.98 LoyaltyProgramLevels

The LoyaltyPrograms table contains the definitions of all configured Loyalty Program Levels.

Column	Type	Allow Nulls	Description
LoyaltyProgramLevelID	Int	N	Primary key, always unique
LoyaltyProgramID	Int	N	Foreign key link to Loyalty Programs table
Name	VarChar(20)	N	Name of the loyalty program level
Description	VarChar(255)	N	Description of the loyalty program level
MinimumPoints	Float	N	Minimum points needed to achieve the level

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyProgramLevelID	P	LoyaltyProgramLevelID	Primary Key

### 3.99 LoyaltyPrograms

The LoyaltyPrograms table contains the definitions of all supported Loyalty Programs.

Column	Type	Allow Nulls	Description
LoyaltyProgramID	Int	N	Primary key, always unique
Name	VarChar(50)	N	Name of the loyalty program
Description	VarChar(100)	N	Description of the loyalty program
MediaID	Int	N	Media definition used to recognize swipes/scans for this loyalty program, also used to generate account numbers. FK to Media table.
Active	Bit	N	Defines whether or not the loyalty program is active or not. Default is active (1).
EffectiveDate	DateTime	N	Date/Time this programs takes effect (starts)
ExpirationDate	DateTime	N	Date/Time this programs expires (ends)
PointExpirationMethod	Int	N	Defines how the points on this loyalty program expire <sup>1</sup>
NeverExpires	Bit	N	Determines of points never expire.
PointExpirationDate	DateTime	Y	Used when the PointExpirationMethod column is set to 2 (points expire on a specified calendar date) to calculate a point expiration date based on a specified date.
PointExpirationDays	Int	Y	Used when the PointExpirationMethod column is set to 1 (points are valid for a specified time period) to calculate a point expiration date based on the date the points were accrued. The GUI enforces a valid range of 0-999999 days for this column.
PrintTktSet	Int	Y	Defines the ticket set to use when printing media for this loyalty program.
TktSetPrinter	Int	Y	Defines the printer to use when printing media for this loyalty program.
UseOnWeb	Bit	Y	Indicates if this loyalty program will be published to and used on the web store.
PluginID	Int	Y	Unique ID of the loyalty plugin used for this loyalty program. 0 indicates that the program is a Gateway loyalty program. If greater than 0, this is a foreign key to Plugins.PluginID.
LoyaltyProgramGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyProgramID	P	LoyaltyProgramID	Primary Key

#### <sup>1</sup> PointExpirationMethod Values

Value	Gateway Constant Name	Description
0	LOYALTY_POINTS_NO_EXPIRATION	Points do not expire
1	LOYALTY_POINTS_DATE_RANGE	Points are valid for a specified time period
2	LOYALTY_POINTS_DATE_EXPIRATION	Points expire on a specified calendar date

### 3.100 LoyaltyRedemptionItems

The LoyaltyRedemptionItems table defines what PLUs can be redeemed using points from the loyalty program, and the point value associated to each PLU.

Column	Type	Allow Nulls	Description
LoyaltyRedemptionItemID	Int	N	Primary key, always unique
LoyaltyProgramID	Int	N	FK to LoyaltyPrograms.LoyaltyProgramID
PLU	Char(20)	N	Ticket/Item PLU, FK to Items.PLU
RedemptionPointValue	Float	N	The number of points needed to purchase this PLU
LoyaltyProgramLevelID	Int	Y	Program Level the item is associated with; 0 indicates no assigned level
LoyaltyRedemptionItemGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKLoyaltyRedemptionItemID	P	LoyaltyRedemptionItemID	Primary Key

### 3.101 MinMaxFares

The MinMaxFares table stores minimum and maximum one way and return trip fares by carrier. This table is only used when calculating prices for intercity transportation tickets sold from the eGalaxy Web Store.

#### Columns

Column	Type	Allow Nulls	Description
MinMaxFareID	Int	N	Primary key, always unique
Carrier	NVarChar(4)	Y	Carrier to which these fares apply
MinFareOW	Money	Y	Minimum amount for a one way ticket
MaxFareOW	Money	Y	Maximum amount for a one way ticket
MinFareRT	Money	Y	Minimum amount for a return trip ticket
MaxFareRT	Money	Y	Maximum amount for a return trip ticket

#### Indexes

Name	Kind	Columns	Purpose
PKMinMaxFaresMinMaxFareID	P	MinMaxFareID	Primary key

### 3.102 Modifiers

The Modifiers table contains the definition of a modifier for a food and beverage item.

#### Columns

Column	Type	Allow Nulls	Description
ModifierID	Int	N	Primary key, always unique
Name	NVarChar(30)	Y	Name of the modifier.
Description	NVarChar(200)	Y	Description of the modifier.
ImageData	Image	Y	The modifier image.
PLU	NVarChar(20)	Y	Foreign key to Items.PLU. Indicates the item that is associated to the modifier. This is necessary if the modifier has a price, or has other modifiers that must be chosen when this modifier is selected (nested modifiers). Optional for all other modifiers.
Inactive	Bit	N	Indicates if the modifier can be selected from most picklists. If 1, the modifier cannot be selected.
SpecialInstruction	Bit	Y	Indicates if this modifier is a special instruction modifier. If 1, selecting this modifier will allow the user to enter custom text and a price (if there is an associated item). The entered text will be displayed in the transaction and on the receipt.

#### Indexes

Name	Kind	Columns	Purpose
PKModifierID	P	ModifierID	Primary key

### 3.103 ModifierGroups

The ModifierGroups table contains the definition of a modifier group. A modifier group defines modifiers that are selected together. This table holds general information about the modifier group. The modifiers are associated in the ModifierGroupDetails table.

#### Columns

Column	Type	Allow Nulls	Description
ModifierGroupID	Int	N	Primary key, always unique
Name	NVarChar(30)	Y	Name of the modifier group.
Description	NVarChar(200)	Y	Description of the modifier group.
ImageData	Image	Y	The modifier group image.
Inactive	Bit	N	Indicates if the modifier group can be selected from picklists. If 1, the modifier group cannot be selected from most picklists.
SelectMax	Int	Y	Indicates the maximum number of modifiers in the group that can be selected. A value of 0 indicates that there is no maximum.
SelectMin	Int	Y	Indicates the minimum number of modifiers in the group that can be selected.

#### Indexes

Name	Kind	Columns	Purpose
PKModifierGroupID	P	ModifierGroupID	Primary key

### 3.104 ModifierGroupDetails

The ModifierGroupDetails table defines the associations of modifiers to modifier groups.

#### Columns

Column	Type	Allow Nulls	Description
ModifierGroupDetailID	Int	N	Primary key, always unique
ModifierGroupID	Int	Y	Foreign key to ModifierGroups.ModifierGroupID. Indicates the modifier group that the modifier is associated with.
ModifierID	Int	Y	Foreign key to Modifiers.ModifierID. Indicates the modifier that is associated to the group.
Sequence	Int	Y	Indicates the sequence in which the modifier will be displayed in the group. The first item in the group has a sequence of 0.

#### Indexes

Name	Kind	Columns	Purpose
PKModifierGroupDetailID	P	ModifierGroupDetailID	Primary key
IXModGrpDtlsModifierGroupID	IX	ModifierGroupID	Used to load modifier group details for a specified group.
IXModGrpDtlsModifierID	IX	ModifierID	Used to load modifier group details that a specified modifier belongs to.

### 3.105 MultipleChoices

The survey function allows the user to define thousands of multiple choice answers, each with fifteen different choices. The **MultipleChoices** table contains such data.

#### Columns

Column	Type	Allow Nulls	Description
MchoiceUniqueID	Int	N	Primary key, always unique. System generated.
MultipleChoiceID	Int	N	Multiple choice ID number. Rows with matching values in this column are all part of the same multiple choice set.
MultipleChoiceType	Char(8)	N	User definable multiple choice set name
ResponseNo	Int	N	The response number. There can be a maximum of 15 possible responses for a survey question.
ResponseText	Char(36)	Y	The description of the response.
MultipleChoiceGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKMultipleChoicesMChoiceID	P	MchoiceUniqueID	Primary key.

### 3.106 MultiSiteAccessCodeConfig

This table is used to associates an AccessCode to one or more attractions, and is used by the Galaxy MultiSite module to synch sales and usage data.

#### Columns

Column	Type	Allow Nulls	Description
MultiSiteAccessCodeConfigID	Int	N	Primary key, always unique
GalaxySiteID	Int	N	FK reference to Sites.GalaxySiteID column
AccessCode	Int	N	FK reference to AccessCodes.AccessCode column

#### Indexes

Name	Kind	Columns	Purpose
PKMultiSiteAccessCodeConfigID	P	MultiSiteAccessCodeConfigID	Primary Key

### 3.107 NameSuffixes

This table contains user-defined name suffixes (i.e. II, III, IV, Esq., etc), which are used for Contacts.

#### Columns

Column	Type	Allows Nulls	Description
NameSuffixID	Int	N	Primary key, always unique
NameSuffix	VarChar(50)	Y	The suffix (i.e. II, III, IV, Esq., etc)
Sequence	Int	Y	This field is used to store an integer value which represents display sequence when displaying name suffixes on Edit Pass form.
Inactive	Bit	Y	This field is the Attribute Active. 1 = Inactive, 0 = Active
NameSuffixGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKNameSuffixID	P	NameSuffixID	Primary Key.
IXNameSuffixesInactive	IX	Inactive	Index to enhance lookup filtering by active NameSuffixes.

### 3.108 NameTitles

This table contains user-defined name titles (i.e. Mr., Mrs., Dr. etc), which are used for Contacts.

#### Columns

Column	Type	Allows Nulls	Description
NameTitleID	Int	N	Primary key, always unique
NameTitle	VarChar(50)	Y	The title (i.e. Mr., Mrs., Dr. etc)
Sequence	Int	Y	This field is used to store an integer value which represents display sequence when displaying name titles on Edit Pass form.
Inactive	Bit	Y	This field is the Attribute Active, 1 = Inactive, 0 = Active
NameTitleGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKNameTitleID	P	NameTitleID	Primary Key.
IXNameTitlesInactive	IX	Inactive	Index to enhance lookup filtering by active NameTitles.

### 3.109 NodeDataConnections

This table is used to associate data to specific nodes. Each piece of data associated to a node will have a row in this table. For example, a certain item may only be useable by a group of nodes. Each of those nodes would have a row in this table associating it to the specific item.

#### Columns

Column	Type	Allows Nulls	Description
NodeDataConnectionID	Int	N	Primary key, always unique
NodeID	Int	N	The node that this data element belongs to
TableID	Int	N	A specific table that this row is associated to. The row in this table is specified by ConnectionID.
ConnectionID	Int	N	The FK reference to the data row. For example, if TableID=8, this row would correspond to the row in Items where ItemID = ConnectionID.

#### Indexes

Name	Kind	Columns	Purpose
PK_NodeDataConnectionID	P	NodeDataConnectionID	Primary Key.

### 3.110 NodeDataStatus

This table is used for store the state of data downloads for each node. It will keep track of the number of publishes that were applied for this node.

#### Columns

Column	Type	Allows Nulls	Description
NodeDataStatusID	Int	No	Primary key, always unique
DownloadNodeID	Int	No	The node number of a station.
ApplyCount	Int	Yes	The number of times a node has downloaded data
LastDownload	Datetime	Yes	The date and time this node downloaded data - updated each time new data is polled. This value will be updated even if there is not new data to download.
LastDBConnection	DateTime	Yes	
JnlRecordCount	Int	Yes	Total number of local sales journal records for this node.
RefreshData	Bit	Yes	When set will cause complete download from central database to local data.
JnlPostedCount	Int	Yes	The total number of local sales journal records that have been posted to the central database.
TicketRecordCount	Int	Yes	Total number of Ticket Sender records for this node.
TicketPostedCount	Int	Yes	The total number of Ticket Sender records that have been posted to the central database.
UpsellState	Int	Yes	Indicates the upsell state for the node. This will determine whether upsell prompting is turned on or off for the node. <sup>1</sup>
CurrentVersion	NVarChar(50)	Yes	The version this node is running when it last checked in with the database.
FailedDownload	Bit	Yes	
NodeStatus	Integer	Yes	Indicates if the node is Active (0) or Inactive (1).
LicenseSerial	NVarChar(100)	Yes	The serial number for the software license.
Signature	NVarChar(max)	Yes	Encrypted signature to determine if data has been modified outside Galaxy

#### Indexes

Name	Kind	Columns	Purpose
PK NodeDataStatusID	P	NodeDataStatusID	Primary Key.
IXNodeDataStatusDownloadNode	IX	DownloadNodeID	This allows the system to select and update a unique row for a given node

<sup>1</sup> UpsellState Values

Value	Gateway Constant Name	Description
0	UPSELL_STATE_ON	Upsell prompting is on.
1	UPSELL_STATE_OFF	Upsell prompting is off.

### 3.11 Nodes

This table contains a list of Galaxy POS stations (nodes).

#### Columns

Column	Type	Allow Nulls	Description
NodeID	Int	N	Primary key, always unique
NodeNumber	Int	N	Alternate key, always unique
PortNumber	Int	Y	The port number
PortDataBits	Int	Y	The data bits for the port used
PortStopBits	Int	Y	The stop bits for the port used
PortBaud	Int	Y	The baud of the port used
PortParity	Int	Y	The parity for the port used
Phone	VarChar(24)	Y	The phone number for this node
Name	VarChar(24)	N	The name of this node
Agency	Int	Y	Agency ID
Window	Int	Y	Window ID
Kind <sup>1</sup>	Int	Y	The kind of node this is (POS, Admission, etc.) <sup>1</sup>
FacilityID	Int	Y	The facility this node belongs to
Description	VarChar(100)	Y	Description of the node
Active	Bit	Y	Is this node inactive?
PersistenceState <sup>2</sup>	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>2</sup>
TCPHost	VarChar(63)	Y	Hostname of the computer running the application with the associated node number. This is limited to 63 characters because of operating system limitations.
TCPPort	Int	Y	Port number that the application's TCP server should listen on.
DeviceType <sup>3</sup>	Int	Y	Represents the type of device this node is. For Nodes, this value should always be 3 (dtNetwork). <sup>3</sup>
DeviceNumber	Int	Y	Represents the numbered instance of the DeviceType in the List of Devices. For example, if Galaxy has NetBIOS as the only network defined in the Device Manager, and all nodes are using NETBIOS, then the node's DeviceNumber values will be 1. If Galaxy has both NETBIOS and TCP/IP (in that order), then all nodes using NETBIOS would have a DeviceNumber value of 1, and all nodes using TCP/IP would have a value of 2. So the value is incremented every time we find a device of the same type (dtNetwork in this case).
NodeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKNodesNodeID	P	NodeID	Primary Key.
AKNodesNodeNumber	A	NodeNumber	Alternate Key.

#### <sup>1</sup> Node Kind Values

Gateway Constant Name	Value	Description
NO_NODE	0	No node
POS_NODE	1	Point of Sale Node
LEAD_NODE	2	Lead Node
ADMISSIONS_NODE	3	Admissions Node
BCAM_NODE	4	BCAM Node
REVENUE_NODE	5	Revenue Node
GROUP_NODE	6	Group Node
DOS_HQ_NODE	7	HQ Concentrator Node
ACCESS_NODE	8	Access Node
DOS_AC52_NODE	9	DOS AC52 Node
DOS_BCAM_NODE	10	DOS BCAM Node
RESELLER_NODE	11	Reseller Node

#### <sup>2</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore Node records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store Nodes.

#### <sup>3</sup> DeviceType Values

Gateway Constant Name	Value	Description
dtUnknown	0	Device type unknown
dtUnsupported	1	Device type unsupported

dtPrinter	2	Device type printer
dtNetwork	3	Device type network
dtMSR	4	Device type MSR
dtCustomerDisplay	5	Device type customer display
dtOther	6	Device type other
dtPad	7	Device type pad
dtScanner	8	Device type scanner
dtTerminal	9	Device type terminal
dtCashDrawer	10	Device type cash drawer
dtTurnstile	11	Device type turnstile
dtSigCap	12	Device type signature capture
dtBarrierGate	13	Device type barrier gate
dtBiometric	14	Device type biometric
dtSmartCard	15	Device type smart card
dtPaymentTerminal	16	Device type payment terminal

### 3.112 NodeSerialNumbers

This table is used to store the association of a node to its serial number used for licensing purposes.

#### Columns

Column	Type	Allow Nulls	Description
NodeSerialNumberID	Int	N	Primary key, always unique
NodeNumber	Int	N	This is a reference to Nodes.NodeNumber
SerialNumber	NVarChar(20)	Y	The serial number that is allocated to the NodeNumber

#### Indexes

Name	Kind	Columns	Purpose
PKNodeSerialNumberID	P	NodeSerialNumberID	Primary Key

### 3.113 NodeVersionHistory

This table tracks the history of versions per node.

#### Columns

Column	Type	Allow Nulls	Description
NodeVersionHistoryID	Int	N	Primary key, always unique
EffectiveDate	DateTime	N	The date and time a version change occurred. The recorded time will be determined when the node checks in with the database, not when the version is installed on the local workstation.
PreviousVersion	NVarChar(50)	Y	The version this node was running before a version change.
CurrentVersion	NVarChar(50)	N	The version the node is/was running on the effective date.

#### Indexes

Name	Kind	Columns	Purpose
PKNodeVersionHistoryID	P	NodeVersionHistoryID	Primary Key

### 3.114 OnlineMenus

The system allows the users to assign a system defined functions to each POS menu buttons. The **OnlineMenu** table contains these menu button function definitions.

#### Columns

Column	Type	Allow Nulls	Description
OnlineMenuID	Int	N	Primary key. Always unique. System generated.
MenuNo	Int	N	A system defined menu level number. <sup>1</sup>
FkeyNo	Int	N	A keyboard short-cut for menu button. The values are between 1 and 8.
MenuFunction	Int	N	The system function number attached to the menu button.
ItemLabel	Char(8)	Y	The label of the menu button.

#### Indexes

Name	Kind	Columns	Purpose
PKOnlineMenusMenuID	P	OnlineMenuID	Primary key.

#### <sup>1</sup> MenuNo Values

Value	Description
1	Ticketing menu 1
2	Ticketing menu 2
3	Ticketing menu 3
4	Forms of Payment menu 2
5	Order Entry menu 1
6	Order Entry menu 2

#

### 3.115 OrcaBlockedCards

The OrcaBlockedCards table is updated via a RESTful endpoint that retrieves the list of Orca blocked cards ("hotlist").

#### Columns

Column	Type	Allows Nulls	Description
OrcaBlockedCardID	Int	No	Primary key, always unique, Identity
FareMediaID	NVarChar(16)	No	The FareMediaID of the Orca Card
FareMediaType	Int	No	The FareMediaType of the Orca Card <sup>1</sup>
BlockTimestamp	DateTime	Yes	The date and time when the Orca Card was placed on the blocked card list.

#### Indexes

Name	Kind	Columns	Purpose
PKOrcaBlockedCards	P	OrcaBlockedCardID	Primary Key.
IXOrcaBlockedCardsFareMedia	IX	FareMediaID, FareMediaType	Unique
IXOrcaBlockedCardsLastUpdate	IX	LastUpdate	

<sup>1</sup> FareMediaType Values

Value	Description
215	Closed Loop
217	Open Loop
220	Barcode
224	Google Virtual Card
225	Apple Virtual Card
226	UID Card
227	External Barcode

### 3.116 OrcaCardPublicKey

This table is used to store the public keys that can be used to validate information stored on an Orca Card.

#### Columns

Column	Type	Allows Nulls	Description
OrcaCardPublicKeyID	Int	No	Primary key, always unique, Identity
KeyIndex	Int	No	Index that is read from the card. The index identifies which public key to use to validate the information on the card
PublicKey	NVarChar(256)	Yes	The public key to use
ValidFrom	Datetime	Yes	The date and time after which the public key will be valid.
KeyVersion	Int	Yes	Version of the public key.
Inactive	Bit	Yes	Used if the public key is no longer in use.

#### Indexes

Name	Kind	Columns	Purpose
PKOrcaCardPublicKeys	P	OrcaCardPublicKeyID	Primary Key.
IXKeyIndexPublicKeyValidFrom	IX	KeyIndex, Inactive, ValidFrom	This allows the system to select valid public keys based on KeyIndex.

#

### 3.117 OrcaResultTypes

This table is used to store the default and GTS override responses to ORCA card transactions.

#### Columns

Column	Type	Allows Nulls	Description
OrcaResultTypeID	Int	No	Primary key, always unique, Identity
OrcaResultType	Int	No	Enumeration type for different result sets
OrcaValue	Int	No	Actual value code returned from ORCA transaction
OrcaName	NVarChar(41)	No	English key from ORCA API
OrcaDefault	NVarChar(75)	Yes	The default interpretation response from ORCA
GTSOverride	NVarChar(75)	Yes	The optional GTS user defined response to OrcaValue
Modifiable	Int	Yes	Flag allowing the record to be editable
Inactive	Int	No	Used if the result type is no longer in use.

#### Indexes

Name	Kind	Columns	Purpose
PKOrcaResultTypes	P	OrcaResultTypesID	Primary Key.
IXOrcaResultTypesResultType	IX	OrcaName, OrcaResultType	This allows the system to select valid Orca results based on OrcaName.

### 3.118 Packages

The Packages table contains the definition of a package.

#### Columns

Column	Type	Allow Nulls	Description
PackageID	Integer	No	Primary key, always unique
PackagePLU	Char(20)	No	PLU of Package Item, Foreign Key to Items Table, always unique
MediaType	Integer	No	Indicates TicketSet(s) to print
PriceMethod	Integer	No	Indicates how the Package Price is determined
UseDetailAccount	Bit	Yes	Determines if the package details use the account values from the package item. If this value is false, the items use their own account values.
PreventPartialReturn	Bit	Yes	Indicates if partial returns are prevented for this type of package.
AllowPartialUpgrade	Bit	Yes	Indicates if partial upgrades are allowed for this type of package.
EnforcePackageEventWindow	Bit	Yes	Indicates if the PackageEventWindow should be enforced on this package.
PackageEventWindow	Integer	Yes	The maximum number of days that is allowed between the first and last events in a package.
PackageGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
AKPackagesPackagePLU	A	PackagePLU	Alternate Key, must be unique.

#### <sup>1</sup> MediaType Values

Value	Gateway Constant Name	Description
0	SINGLE_MEDIA	Package Item ticket set (if any) will print
1	INDIVIDUAL_MEDIA	Each Package Detail ticket set (if any) will print
2	MIXED_MEDIA	Package Item ticket set (if any) will print; and each Package Detail ticket set (if any) will print

#### <sup>1</sup> PriceMethod Values

Value	Gateway Constant Name	Description
0	METHOD_PKG_ITEM_PRICE	Price is that entered on the Accounting Tab of the Package Item (unless overridden by price schedule price on Package Item)
1	METHOD_SUM_OF_DETAILS	Price of the Package is the sum of the current prices of all of the package details.

### 3.119 PackageDetails

The PackageDetails table contains the detailed information associated to a package.

#### Columns

Column	Type	Allow Nulls	Description
PackageDetailID	Integer	No	Primary key, always unique
PackagePLU	Char(20)	No	PLU of master Package Item, Foreign Key to Packages table, always unique in combination with PackageSeq
PackageSeq	Integer	No	Indicates the ordinal position of the item within the package, always unique in combination with PackagePLU
ItemPLU	Char(20)	No	PLU of the detail, Foreign Key to Items table
ItemPkgPrice	Float	Yes	Package price for qty of 1 - could be dollar amount or percentage (If package pricing method is sum_of_details or if PriceBasis is Remainder, this can be null.)
PriceBasis	Int	Yes	Basis of Package Detail Price <sup>1</sup> - (if package pricing method is sum_of_details, this can be null - price comes from Item table)
Quantity	Integer	No	Quantity for the ItemPLU included in the Package
Discountable	Boolean	No	If not set, this package detail cannot be discounted when the Package is discounted
PrintTktSet	Boolean	Yes	In the case of INDIVIDUAL_MEDIA or MIXED_MEDIA, this value determines if ticket(s) print for this detail.
TicketSetID	Integer	Yes	If populated, this ticket set overrides the ticket set on the Item record for this detail.
IssuanceType	Integer	Yes	Determines how the PLU will be issued. Example: issue at the time of the sale, issue at a later time. <sup>2</sup>
TktSetPrinter	Integer	Yes	The printer to print to when printing the ticket set.
RestrictSubsequentEventSelection	Bit	Y	Boolean value that determines if subsequent events are allowed to be selected or not
MinimumOffsetMinutes	Int	Y	Minimum offset used to link events
MaximumOffsetMinutes	Int	Y	Maximum offset used to link events
OffsetCalculationMethod	Int	Y	Method used to calculate the offset minutes  See Type Values below <sup>3</sup>
FirstEvent	Bit	Y	Determines if the event detail is required to be the first chronological event in the package.
EventEndsEarlyMinutes	Integer	Y	The number of minutes that the event lets out before the actual event end time. When this value is set to a non-zero value, the time calculations for minimum and maximum offset are based on this time instead of the end time on the event. This will typically be used if the event times include cleanup time that does not require the guests to stay until the end.
EnforceMinimumOffset	Bit	Y	Determines if the minimum offset is enforced for the event package detail when selling the package. If this value is 0, events may overlap this event.
EnforceMaximumOffset	Bit	Y	Determines if the maximum offset is enforced for the event package detail when selling the package. If this value is 0, there is no restriction on the amount of time between other events in the package.
OverrideGiftAid	Bit	Y	If set, indicates that the Gift Aid configured on the package detail overrides the Gift Aid configured on the item.
GiftAidType	Int	Y	The Gift Aid type for the package detail that overrides the type on the item. Only used if OverrideGiftAid is set. <sup>4</sup>
GiftAidAmount	Money	Y	The amount of GiftAid for the package detail. When the GiftAid type is percentage, this is a percentage value. Only used if OverrideGiftAid is set.
UpgradeValue	Money	Y	When UpgradeValueType is set to PACKAGE_DETAIL_UPGRADE_VALUE (0), this value will be used as the package details upgrade value. A value of 0 means the sale price will be used.
UpgradeValueType	Integer	Y	Determines which upgrade value should be used for the package detail upgrade value. <sup>5</sup>
PackageDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPackageDtlsPackageDetailID	P	PackageDetailID	Primary Key

#### 1 PriceBasis Values

Value	Gateway Constant Name	Description
1	PKGDTL_ACTUAL_DOLLAR	Detail price is an actual dollar amount entered by user
2	PKGDTL_PERCENTAGE	Detail price is a percentage of the Package Item Price
3	PKGDTL_REMAINDER	Detail price is the amount remaining from the Package Item Price after all other details are priced (can be assigned to only one detail in a Package)

#### 2 IssuanceType Values

Value	Gateway Constant Name	Description
0	ISSUANCE_TYPE_SALE	The package detail will be issued at the time of the sale.
1	ISSUANCE_TYPE_DEFERRED	The package detail will be issued at a later time.

#### 3 OffsetCalculationMethod Values

Value	Description
0	Calculate offset using start time of the event
1	Calculate offset using end time of the event

#### 4 GiftAidType Values

Value	Gateway Constant Name	Description
0	gaNone	No GiftAid allowed for the item.
1	gaFixedAmount	Fixed amount of Gift Aid to add to the item.
2	gaFull	Full ticket price can be used as GiftAid.

5 | gaPercentage | Calculate Gift Aid amount as a percentage of the item price |

### 5 UpgradeValueType Values

Value	Gateway Constant Name	Description
0	ITEM_UPGRADE_VALUE	Use upgrade value defined on the item.
1	PACKAGE_DETAIL_UPGRADE_VALUE	Use upgrade value defined on the package detail. When this value is 0, use the item's sale price.
2	EXCLUDE_UPGRADE_VALUE	Do not include an upgrade value for this detail.

### 3.120 Pages

This table is used in the maintenance of Lead pages.

#### Columns

Column	Type	Allow Nulls	Description
PageUniqueID	Integer	No	Primary key, always unique
PageID	Integer	No	Local Key
Label	Varchar(80)	Yes	Name or short description
Message	Varchar(80)	Yes	The page message

#### Indexes

Name	Kind	Columns	Purpose
PKPageUniqueID	P	PageUniqueID	Primary key, must be unique.

### 3.121 PaymentContractItems

#### Columns

Column	Type	Allow Nulls	Description
PaymentContractItemID	Integer	N	Primary Key, unique number.
AuxID	Integer	Y	OrderLineID for tickets, passes, and items. NoteID from Notes table.
ContractItemType	Integer	Y	Item, Pass, or Note <sup>1</sup>
PaymentContractID	Integer	Y	Link to Payment Contract table
MinimumDownPayment	Money	Y	The minimum required down payment
VariablePayment	Money	Y	Variable Payment amount for the individual contract item based on the plan criteria.
AdditionalPayment	Money	Y	Records the amount of money required at the time of sale that is separate from the down payment. Typically this amount will be tied to stored value.

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentContractItemsPCItemID	P	PaymentContractItemID	Primary Key.
IXPCIContractIDLastUpdate	IX	PaymentContractID, LastUpdate	Used by the Contract Processor export process to search for changes to these tables
IXContractItemType	IX	ContractItemType	Improve query performance
IXAuxID	IX	AuxID	Improve query performance

<sup>1</sup> ContractItemType Values

Value	Gateway Constant Name	Description
0	PASS_CONTRACT_ITEM_TYPE	Pass
1	TICKET_CONTRACT_ITEM_TYPE	Ticket
2	ITEM_CONTRACT_ITEM_TYPE	Item
3	NOTE_CONTRACT_ITEM_TYPE	Note
4	FEE_CONTRACT_ITEM_TYPE	Fee
5	PACKAGEONPASS_CONTRACT_ITEM_TYPE	Package on Pass
6	ORDER_FEE_CONTRACT_ITEM_TYPE	Order Fee

### 3.122 PaymentContracts

#### Columns

Column	Type	Allow Nulls	Description
PaymentContractID	Integer	N	Unique number to identify this payment agreement.
ContactID	Integer	Y	Link to CustContactID in CustContact table
PlanID	Integer	Y	Link to payment plan in PaymentPlan table
RenewContract	Bit	Y	1 = The plan renews. 0 = The plan does not renew.
PaymentContractStatusID	Integer	Y	Status Code or Status Code Group ID for the current status of this payment contract.
PaymentContractEffective	DateTime	Y	Effective date of payment plan.
PaymentStart	DateTime	Y	Date first payment is due, calculated based on criteria on plan.
NextPaymentDue	DateTime	Y	Date of next scheduled payment.
PaymentContractRenew	DateTime	Y	Date plan should renew, calculated based on criteria on plan.
DownPaymentAmount	Money	Y	Dollar amount of first payment, based on the criteria on plan.
PaymentAmount	Money	Y	Dollar amount of regular payment.
RemainingPayments	Integer	Y	Number of remaining payments left on payment contract.
CurrentBalance	Money	Y	Current balance due on payment contract.
Arrears	Money	Y	Dollar amount of missed payments.
OrderID	Integer	Y	Order ID to link contract to Order.
ContractCancelDate	DateTime	Y	Cancellation will be used to end a payment contract prior to the expiration date.
CCDeclineDate	DateTime	Y	Date credit card was declined
WelcomeWebTemplateID	Integer	Y	Template ID if different from one on plan.
DeclineWebTemplateID	Integer	Y	Template ID if different from one on plan.
ContractWebTemplateID	Integer	Y	Template ID if different from one on plan.
ConvertPlanWebTemplateID	Integer	Y	Template ID if different from one on plan.
ChangeCCWebTemplateID	Integer	Y	Template ID if different from one on plan.
CCExpireWebTemplateID	Integer	Y	Template ID if different from one on plan.
RenewalWebTemplateID	Integer	Y	Template ID if different from one on plan.
BillingCycles	Integer	Y	Number of billing cycles, calculated based on payment plan duration, first payment offset and length of cycle.
RecurrencePatternID	Integer	Y	Link to billing recurrence pattern
ContactPaymentInfoID	Integer	Y	Link to contact payment info table for credit card information.
PreviousContractID	Integer	Y	Link to PaymentContract table for previous Contract.
SuspendWebTemplateID	Integer	Y	Template ID if different from one on plan.
MissedPayments	Integer	Y	Number of consecutive missed payments. Incremented every time a regular payment fails. Reset to 0 every time a successful payment is made.
ContactMethod <sup>1</sup>	Integer	Y	Method by which a contact will receive statements generated automatically by the automated processes. This can be either Mail or Email at the moment.
ContractTotal	Money	N	Total amount of contract at activation time
ExportDate	DateTime	Y	The date and time that the contract was exported using the Contract Processor export process.
LastModifiedExportDate	DateTime	Y	The last date and time that the contract was exported as a modified contract using the Contract Processor export process.
VariablePaymentAmount	Money	Y	Dollar amount of variable payment, based on the criteria on plan.
VariablePaymentDate	DateTime	Y	Date the variable payment will be charged based on the criteria on plan.
VariablePaymentCompleted	Bit	Y	Set to 1 when Variable Payment has been paid. Can be ignored if VariablePaymentAmount is equal to zero.
AdditionalPaymentAmount	Money	Y	The portion of the DownPaymentAmount that was required because of the AdditionalPayment on one or more contract items.

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentContractsPyntCntrctID	P	PaymentContractID	Primary Key.
IXP CPC CID C P II DBal LastUpdate	IX	PaymentContractID, ContactID, ContactPaymentInfoID, CurrentBalance, LastUpdate	Used by the Contract Processor export process to search for changes to these tables
IXPaymentContractsOrderID	IX	OrderID	Index to speed up query to load contracts by order ID. This is used by customer payments to determine how much of the current balance is due to contracts.
IXPaymentContractsExportDate	IX	ExportDate	Index to speed up searches on export date
IXPaymentContractsLMEDate	IX	LastModifiedExportDate	Index to speed up searches on last modified export date

<sup>1</sup> ContactMethod Values

Value	Gateway Constant Name	Description
0	EMAIL_CONTRACT_CONTACT_METHOD	Contact will receive statements by email
1	MAIL_CONTRACT_CONTACT_METHOD	Contact will receive statements by mail

### 3.123 PaymentContractStatuses

#### Columns

Column	Type	Allow Nulls	Description
PaymentContractStatusID	Integer	N	Unique number to identify status code.
Name	Varchar(50)	Y	Name of status.
Description	Varchar(256)	Y	Description of status
Active	Bit	Y	1 = Active, valid for use. 0 = Inactive, invalid for use.
StatusAccess	Integer	Y	0 = Allow. 1 = Block.
Kind <sup>1</sup>	Integer	Y	Map to constant system codes used by automated processes.
PaymentContractStatusGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentContractStatusesPCSID	P	PaymentContractStatusID	Primary Key.

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
0	CONTRACT_STATUS_KIND_INACTIVE	Inactive
1	CONTRACT_STATUS_KIND_ACTIVE	Active
2	CONTRACT_STATUS_KIND_DECLINED	Declined
3	CONTRACT_STATUS_KIND_SUSPENDED	Suspended
4	CONTRACT_STATUS_KIND_PAID	Paid
5	CONTRACT_STATUS_KIND_CANCELED	Canceled
6	CONTRACT_STATUS_KIND_CLOSED	Closed
7	CONTRACT_STATUS_KIND_VOIDED	Voided

### 3.124 PaymentPlanConfig

A configuration table for Payment Plans

#### Columns

Column	Type	Allow Nulls	Description
PaymentPlanConfigID	Integer	N	Primary key, always unique.
PaymentContractMax	Integer	Y	Payment contracts one contact may have 0 = unlimited.
PlanPassMax	Integer	Y	Maximum number of passes per plan.
AllowMultipleContractsInOrderEntry	Bit	Y	Allow Multiple Contracts In Order Entry.
PaymentContractFOP	Integer	Y	FOP ID for charge account to use to balance contract transactions.
PaymentSuspend	Integer	Y	Number of missed payments to put contracts in a suspend status. 0 = No limit.
PaymentCancel	Integer	Y	Number of missed payments to put contracts in a cancel status. 0 = No limit.
PrintAtPOS	Bit	Y	1 = Print, 0 = No Print.
ProcessArrears	Bit	Y	Process arrears with Regular payments.
RenewArrears	Bit	Y	Renew contracts with an arrears balance.
CombineCCPayments	Bit	Y	Combine payments by credit card number so only one payment type is processed per cc number.
ActiveStatusCode	Integer	Y	PaymentContractStatusID of the system's current active status.
InactiveStatusCode	Integer	Y	PaymentContractStatusID of the system's current inactive status.
DeclinedStatusCode	Integer	Y	PaymentContractStatusID of the system's current declined status.
SuspendedStatusCode	Integer	Y	PaymentContractStatusID of the system's current suspended status.
PaidStatusCode	Integer	Y	PaymentContractStatusID of the system's current paid status.
CanceledStatusCode	Integer	Y	PaymentContractStatusID of the system's current canceled status.
ClosedStatusCode	Integer	Y	PaymentContractStatusID of the system's current closed status.
EmailSenderName	Text	Y	Name of the email sender for contract statements emails.
EmailSenderAddress	Text	Y	Email address in the From field of contract statement emails
ReplyToEmailAddress	Text	Y	Email address that replies should be sent to. If blank, replies will go to sender address.
PromptForOwnerAtBeginning	Bit	Y	Determines if we prompt for a contract owner before beginning the payment calculator wizard (True), or if we select the contract owner at the summary step (False)
AllowMultipleDownPayments	Bit	Y	Used to determine if multiple payments are allowed to satisfy the down payment (True), or if the down payment must be satisfied by a single payment (False)
AlwaysConfirmCCForRecurring	Bit	Y	Used to determine if the user is always prompted to select a credit card for the recurring payment (True), or if the credit card used as the down payment is automatically used (False).
AutoIssueContractItems	Bit	Y	Used to determine if the contract items are issued after the confirmation screen of the wizard (True), or if the option to issue is presented on the payment screen (False)
AutoViewContractStatement	Bit	Y	Used to determine if the contract statement is automatically displayed (True), or if the Statement screen is displayed with the option to preview or print any of the contract statements (False)
VerifyContractOwner	Bit	Y	Used to determine if the contract owner confirmation screen is displayed or not. This option is only available when PromptForOwnerAtBeginning option is FALSE.
VoidedStatusCode	Integer	Y	VoidedContractStatusID of the system's current voided status
EnableTestAuthorizeRecurring	Bit	Y	Determines if the credit card used for the recurring payment in a payment contract will be verified with an authorization to test the validity of the account before the contract can be accepted. The authorization will only occur if the protocol supports it, and if the card used for the recurring payment is different than the form of payment used for the down payment. This option is limited to protocols that support the option (Initially this will only be Stratus).
RecurringTestAuthorizeAmount	Money	Y	The amount to be used for the test authorization. The amount to use should be an amount deemed to be acceptable for a test authorization for account verification. The test is an actual authorization, but will not be journalized, so the amount will not be settled and charged to the customer. The amount could create a temporary hold on the account, so it should be small enough that it will not affect the customer's account and prevent other purchases with the card. Typically \$0 or \$1 would be an acceptable test amount.
ContactFieldAttributeGroupID	Int	Y	A reference back to the FieldAttributeGroups table. This defines what field attributes will be defined for the contract owner.
ReturnPLU	Char(20)	Y	PLU to use when voiding a contract and a ticket/pass was used for the down payment
MinimumContractOwnerAge	Integer	Y	Minimum contract owner age for all payment plans.
AutoViewWelcomeStatement	Bit	Y	Used to determine if the welcome letter statement will automatically display when the guest accepts the contract. This option is only available when AutoViewContractStatement is True.
PaymentCalculatorCustomerID	Integer	Y	Indicates the ID of the customer that will always be used when selling a payment contract through the Payment Calculator. If this value is 0 or NULL, the customer must be selected at time of sale. Foreign key to Customers.CustomerID.
MarkContractGiftAsPledge	Bit	Y	Indicates if gifts for payment contract items should be marked with a gift type of "Pledge". If the gifts are marked as "Pledge", making payments against the contract will create "Cash" type gifts.
LimitPlansByPassExpiration	Bit	Y	1 = Don't allow payment contracts to be created for passes that would expire before payment contract is paid in full.
AllowContactlessRecurringPayment	Bit	Y	1 = Allow contactless payment methods to be used for recurring payments.
PaymentPlanConfigGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentPlanConfigPmtPlnCfgID	P	PaymentPlanConfigID	Primary Key.

### 3.125 PaymentPlanLog

#### Columns

Column	Type	Allow Nulls	Description
PaymentPlanLogID	Integer	N	Primary Key, unique number.
AuxTableID	Integer	Y	Link to PaymentPlan or PaymentContract table.
LogType <sup>1</sup>	Integer	Y	This determines the table that this record links to in the AuxTableID field. This can be PaymentPlan or PaymentContract.
LogDateTime	DateTime	Y	Date Time stamp of when note was entered.
UserID	Integer	Y	ID of user who entered the note.
Node	Integer	Y	Node where activity occurred.
Activity	Text	Y	Log contract change.
OrderlineID	Integer	Y	ID linking to the Orderline table (PaymentContract only).
AmountOfPayment	Money	Y	Amount of payment (PaymentContract only).
Date	DateTime	Y	Date of payment (PaymentContract only).
FOP	Integer	Y	Form of payment used (PaymentContract only).
ActivityType <sup>2</sup>	Int	Y	Type of activity for this log record
ActivityStatus <sup>3</sup>	Bit	Y	Result of activity
ActivityTriggerType <sup>4</sup>	Int	Y	Indicates how the activity was triggered.

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentPlanLogPmtPlanLogID	P	PaymentPlanLogID	Primary Key.
UCXPaymentPlanLogDateID	UC	LogDateTime, PaymentPlanLogID	Unique Clustered Index to speed up the record search by date range. Payment Contract Reports.
IXPaymentPlanDateLogTypeOLID		Date, LogType, OrderLineID	Index used in Payment Contract Report
IXPaymentPlanLogAuxActTrigType		LogType, AuxTableID, ActivityTriggerType	Index used in Contract Processor to gather recurring payment counts.

#### <sup>1</sup> LogType Values

Value	Gateway Constant Name	Description
0	PAYMENT_PLAN_LOG_TYPE	Log entry is for a payment plan
1	PAYMENT_CONTRACT_LOG_TYPE	Log entry is for a payment contract

#### <sup>2</sup> ActivityType Values (works in conjunction with LogType and ActivityStatus)

Value	Gateway Constant Name	Description
1	PLAN_MODIFIED_ACTIVITY_TYPE	Payment Plan was modified (when LogType = 0)
1	CONTRACT_RENEWAL_ACTIVITY_TYPE	Contract renewal (when LogType = 1)
2	CONTRACT_CONVERSION_ACTIVITY_TYPE	Contract conversion (when LogType = 1)
3	CONTRACT_MANUAL_PAYMENT_ACTIVITY_TYPE	Contract Manual Payment (when LogType = 1)
4	CONTRACT_SCHEDULED_PAYMENT_ACTIVITY_TYPE	Contract Scheduled Payment (when LogType = 1)
5	CONTRACT_ARREARS_PAYMENT_ACTIVITY_TYPE	Contract Arrears Payment (when LogType = 1)
6	CONTRACT_DOWN_PAYMENT_ACTIVITY_TYPE	Contract Down Payment (when LogType = 1)
7	CONTRACT_VOID_PAYMENT_ACTIVITY_TYPE	Contract Void Payment (when LogType = 1)
8	CONTRACT_MODIFIED_ACTIVITY_TYPE	Contract modified (when LogType = 1)
9	CONTRACT_ITEM_ADD_ACTIVITY_TYPE	Contract item add (when LogType = 1)
10	CONTRACT_ITEM_UPDATE_ACTIVITY_TYPE	Contract item update (when LogType = 1)
11	CONTRACT_ITEM_REMOVE_ACTIVITY_TYPE	Contract item remove (when LogType = 1)
12	CONTRACT_FULL_PAYMENT_ACTIVITY_TYPE	Contract full payment (when LogType = 1)
13	CONTRACT_EXPORT_PAYMENT_ACTIVITY_TYPE	Contract export payment (when LogType = 1)
14	CONTRACT_RETURN_CREDIT_PAYMENT_ACTIVITY_TYPE	Contract return credit payment (when LogType = 1)
15	CONTRACT_UPGRADE_PAYMENT_ACTIVITY_TYPE	Contract upgrade payment (when LogType = 1)
16	CONTRACT_VOID_CONTRACT_ACTIVITY_TYPE	Contract void contract (when LogType = 1)
17	CONTRACT_DOWN_AND_FEE_PAYMENT_ACTIVITY_TYPE	Contract down and fee payment (when LogType = 1)
18	CONTRACT_RECURRING_TEST_AUTHORIZATION_HOST_OFFLINE_ACTIVITY_TYPE	Test authorization for the recurring payment credit card was skipped due to the host being offline (when LogType = 1)
19	CONTRACT_VARIABLE_PAYMENT_ACTIVITY_TYPE	Contract variable payment (when LogType = 1)

#### <sup>3</sup> ActivityStatus Values

Value	Gateway Constant Name	Description
0		Activity has Failed
1		Activity was Successful

#### <sup>4</sup> ActivityTriggerType Values

Value	Gateway Constant Name	Description
0	MANUAL_ACTIVITY_TRIGGER_TYPE	Activity was manually triggered
1	AUTOMATIC_ACTIVITY_TRIGGER_TYPE	Activity was automatically triggered

### 3.126 PaymentPlanPriceData

This table contains prices for each PLU at each Payment Plan price assigned to the item.

#### Columns

Column	Type	Allow Nulls	Description
PaymentPlanPriceDataID	Int	N	Primary Key, unique
CategoryID	Int	N	The SalesChannelDetailID for the current PLU.
PLU	Char(20)	N	This identifies what PLU the price data is defining.
PaymentPlanID	Int	N	This is the payment plan the price data is defining.
DownPayment	Money	N	The minimum down payment amount for this PLU under this plan.
RecurringPayment	Money	N	The payment required at each interval.
RecurrenceInterval	Int	N	The number of months or days the plan is defined for.
RecurrenceType <sup>1</sup>	Int	N	This determines they kind of interval the plan has. <sup>1</sup>
BillingCycle	Int	N	The number of payments required.
PaymentDurationUnits <sup>2</sup>	Int	N	This determines the data type of the billing cycle. <sup>2</sup>
VariablePayment	Money	Y	The variable payment amount for this PLU under this plan.
VariablePaymentActive	Bit	Y	1 = Variable Payments are enabled, 0 = Variable Payments are disabled.
VariablePaymentOffset	Int	Y	Length of time before variable payment is due.
VariablePaymentOffsetUnits	Int	Y	Units (days/weeks/months) of time for VariablePaymentOffset. <sup>3</sup>
AdditionalPayment	Money	Y	Records the amount of money required at the time of sale that is separate from the down payment. Typically this amount will be tied to stored value.
PaymentPlanPriceDataGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentPlanPriceDataID	P	PaymentPlanPriceDataID	Primary Key.

#### <sup>1</sup> RecurrenceType Values

Value	Gateway Constant Name	Description
0	RECURRENCE_TYPE_NONE	
1	RECURRENCE_TYPE_DAILY	
2	RECURRENCE_TYPE_WEEKLY	
3	RECURRENCE_TYPE_WEEKLY_NTH	
4	RECURRENCE_TYPE_MONTHLY	
5	RECURRENCE_TYPE_MONTHLY_NTH	
6	RECURRENCE_TYPE_YEARLY	

#### <sup>2</sup> PaymentDurationUnits Values

Value	Gateway Constant Name	Description
0	DAYS_TIME_UNIT	
1	WEEKS_TIME_UNIT	
2	MONTHS_TIME_UNIT	
3	YEARS_TIME_UNIT	

#### <sup>3</sup> VariablePaymentOffsetUnits Values

Value	Gateway Constant Name	Description
0	VARIABLEPAYMENT_OFFSETUNITS_DAYS	The units of VariablePaymentOffset is Days
1	VARIABLEPAYMENT_OFFSETUNITS_WEEKS	The units of VariablePaymentOffset is Weeks
2	VARIABLEPAYMENT_OFFSETUNITS_MONTHS	The units of VariablePaymentOffset is Months

### 3.127 PaymentPlanRecurrencePatterns

The PaymentPlanRecurrencePatterns table contains the recurrence pattern data for Payment Plans.

#### Columns

Column	Type	Allow Nulls	Description
PaymentPlanRecurrencePatternID	Integer	N	Primary Key, unique number.
RecurrenceID	Integer	Y	ID used by Btree (Not used currently).
Name	VarChar(50)	Y	Recurrence Name
DayOfMonth	Integer	Y	Day of Month selected. Can be any integer between 1 and 31.
DayOfMonthMask	Integer	Y	Selected days of the month (Not used currently).
DayOfWeekMask	Integer	Y	Stores day(s) of the week (Sunday thru Saturday) (Not used currently).
Instance	Integer	Y	Will be used by Month and Year (Not currently used).
Interval	Integer	Y	(n) number of RecurrenceTypes. If RecurrenceType is Daily, this is the number of days. If Monthly, this is the number of months. Currently, can only be 1 for Monthly.
MonthOfYear	Integer	Y	Selected month of the year (Not currently used).
RecurrenceType	Integer	Y	Indicates the frequency of payments. <sup>1</sup>
StartDate	DateTime	Y	Start date of recurrence.
EndDate	DateTime	Y	Ending date of recurrence.
EnableTime	Bit	Y	Turn on time checking of recurrence (Not currently used).
StartTime	DateTime	Y	Start time of recurrence if time is enabled.
EndTime	DateTime	Y	End time of recurrence if time is enabled (not currently used).
PaymentPlanRecurrencePatternGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentPlanRecurrencePatternID	P	PaymentPlanRecurrencePatternID	Primary Key.

#### <sup>1</sup> RecurrenceType Values

Value	Gateway Constant Name	Description
0	NONE_RECURRENCE_TYPE	No recurrence type
1	DAILY_RECURRENCE_TYPE	Payment Plan payments due daily.
2	WEEKLY_RECURRENCE_TYPE	Payment plan payment due weekly.
3	WEEKLY_NTH_RECURRENCE_TYPE	Payment Plan payment due every n weeks.
4	MONTHLY_RECURRENCE_TYPE	Payment Plan payment due monthly.
5	MONTHLY_NTH_RECURRENCE_TYPE	Payment Plan payment due every n months.
6	YEARLY_RECURRENCE_TYPE	Payment Plan payment due annually.

### 3.128 PaymentPlans

Payment plan definitions.

#### Columns

Column	Type	Allow Nulls	Description
PaymentPlanID	Integer	N	Primary key, always unique.
Active	Bit	Y	1 = Active. 2 = Inactive, invalid for use.
Name	Varchar(50)	Y	Payment plan name.
Description	Varchar(256)	Y	Payment plan description
PaymentDuration	Integer	Y	Length of time between the first and last payment
PaymentDurationUnits	Integer	Y	Units of payment duration time period. Days, Months,
PlanDuration	Integer	Y	The total number of months before the payment plan renews.
RecurrencePatternID	Integer	Y	[ No longer used - Replaced by PaymentPlanRecurrencePatternID ]
RegularPaymentOffset	Integer	Y	Time period until the first payment is due. 0 = Due at time of sale.
OffsetUnit	Integer	Y	Units of duration time period. Days, Months,
DownPaymentAmount	Money	Y	Required amount to be paid at time of purchase.
DownPaymentAmountUnits	Integer	Y	Percent, Dollars.
MaxpassesPaymentContract	Integer	Y	Maximum number of passes that can be added to one contract. 0 = Unlimited.
MinpassesPaymentContract	Integer	Y	Minimum number of passes required for a contract.
WelcomeWebTemplateID	Integer	Y	Statement ID
DeclineWebTemplateID	Integer	Y	Statement ID
ContractWebTemplateID	Integer	Y	Statement ID
ConvertPlanWebTemplateID	Integer	Y	Statement ID
ChangeCCWebTemplateID	Integer	Y	Statement ID
CCExpireWebTemplateID	Integer	Y	Statement ID
CCExpireOffset	Integer	Y	Time period before/after CC expiration to generate statement.
CCExpireUnits	Integer	Y	Days before, Days after, Weeks before, Weeks after, Months before, Months after
RenewalWebTemplateID	Integer	Y	Statement ID
RenewalOffset	Integer	Y	Time period before/after renewal date to generate statement.
RenewalUnits	Integer	Y	Days before, Days after, Weeks before, Weeks after, Months before, Months after
AutoRenew	Bit	Y	Automatically Renew.
RenewPlanID	Integer	Y	Plan ID for plan for auto renewal if different from current plan.
SuspendWebTemplateID	Integer	Y	Statement ID
DownPaymentFOPSetID	int	Y	This is the ID of the FOP group containing the FOPs that will be available for making the down payment on the payment plan. This value is a reference back to FOPSetID in the FOPSets table.
RecurringPaymentFOPSetID	int	Y	This is the ID of the FOP group containing the FOPs that will be available for making recurring payments on the payment plan. This value is a reference back to FOPSetID in the FOPSets table.
PostalCodeRangeID	Int	Y	A reference back to the PostalCodeRanges table. This defines what postal codes will be accepted for this payment plan
PaymentPlanRecurrencePatternID	Int	Y	Link to PaymentPlanRecurrencePatterns table.
VariablePaymentActive	Bit	Y	1 = Variable Payments are enabled, 0 = Variable Payments are disabled
VariablePaymentAmount	Money	Y	Amount (\$ or %) of the variable payment
VariablePaymentAmountUnits	Int	Y	Determines if VariablePaymentAmount is currency or percentage. <sup>1</sup>
VariablePaymentOffset	Int	Y	Length of time before variable payment is due
VariablePaymentOffsetUnits	Int	Y	Units (days/weeks/months) of time for VariablePaymentOffset. <sup>2</sup>
AllowBlankRecurringPayment	Bit	Y	Blank Payments are allowed for PaymentContracts if this value is 1
PaymentPlanGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPaymentPlansPaymentPlanID	P	PaymentPlanID	Primary Key.

#### <sup>1</sup> VariablePaymentAmountUnits Values

Value	Gateway Constant Name	Description
0	VARIABLEPAYMENT_AMOUNTUNITS_AMOUNT	Variable Payment Amount should be treated as a static currency amount
1	VARIABLEPAYMENT_AMOUNTUNITS_PERCENT	Variable Payment Amount should be treated as a percentage of the item/tickets currency amount.

#### <sup>2</sup> VariablePaymentOffsetUnits Values

Value	Gateway Constant Name	Description
0	VARIABLEPAYMENT_OFFSETUNITS_DAYS	The units of VariablePaymentOffset is Days
1	VARIABLEPAYMENT_OFFSETUNITS_WEEKS	The units of VariablePaymentOffset is Weeks
2	VARIABLEPAYMENT_OFFSETUNITS_MONTHS	The units of VariablePaymentOffset is Months

### 3.129 PkgInstances

The PkgInstances table contains information about a package that has been sold. This is where the information not related to access control is stored. Access control information about a package is stored in the SuperTickets table.

#### Columns

Column	Type	Allow Nulls	Description
PkgInstanceID	Integer	No	Unique ID
VisualID	VarChar(40)	No	The visual ID of the package
PLU	VarChar(20)	No	The PLU of the package
NodeNo	Integer	No	Node number of the node that sold the package
TransNo	Integer	No	The transaction number of the package sale transaction
TransDate	DateTime	Yes	The transaction date of the package sale transaction
ContactID	Int	Yes	The contact associated to this package instance. This column is populated whenever a contact is present in the transaction where the package is purchased.
PreviousVisualID	Varchar(40)	Yes	Indicates the visual ID of the previous package when the package has been reprinted or upgraded from a package. When upgraded from a ticket or pass, this is the visual ID from the Tickets or Passes table. This is blank if the package was not created by reprinting or upgrading.
NextVisualID	Varchar(40)	Yes	Indicates the visual ID of the next ticket or pass when the package was upgraded to something other than a package. This allow lineage linking without Super Tickets.
Reprinted	Bit	Yes	Indicates that the package has already been reprinted.
ReprintOnNextScan	Bit	Y	Indicates that the package should be reprinted on next scan at ACS.

#### Indexes

Name	Kind	Columns	Purpose
PKPkgInstancesPkgInstanceID	P	PkgInstanceID	Primary Key.
IXPkgInstancesVisualIDTrans	A	VisualID, NodeNo, TransNo	To allow faster searches on a package sold in a particular transaction
IXPkgInstancesPreviousVisualID	A	PreviousVisualID	Allows faster searches when determining later packages in the lineage.
IXPkgInstancesNextVisualID	A	NextVisualID	Allows faster searches when determining if tickets or passes were upgraded from a package.

### 3.130 PkgInstanceDetails

The PkgInstanceDetails table Contains information about items within a package that has been sold. This is where the information not related to access control is stored. Access control information about a package is stored in the SuperTickets table.

#### Columns

Column	Type	Allow Nulls	Description
PkgInstanceDetailID	Integer	N	Unique ID
PkgInstanceID	Integer	N	The unique ID of the master package instance. This is a foreign key to the PkgInstances table
PackageDetailID	Integer	N	The unique ID of the package detail that this instance was created from. This is a foreign key to the PackageDetails table
VisualID	VarChar(40)	N	The visual ID of the item in the package
PLU	VarChar(20)	N	The PLU of the item in the package
TicketType <sup>1</sup>	Integer	N	The type of item in the package
IssuanceStatus <sup>2</sup>	Integer	N	Status indicating if the entitlement has been issued
Quantity	Integer	N	The quantity of the detail in the package (Used for item details)
Amount	Float	N	The unit price of the detail
Tax	Float	N	The tax (per unit) on the detail
NodeNo	Integer	N	The node number of the sale transaction for the detail
TransNo	Integer	N	The transaction number of the sale transaction for the detail
TransDate	DateTime	Y	The transaction date of the sale transaction for the detail
JnlDetailID	Integer	N	The unique ID of the corresponding journal detail from the sale in the JnlDetails table. NOTE: This field is no longer used.
MediaVisualID	Varchar(40)	N	The visual ID on the media for the detail. This will be the package visual ID for single media packages, and will be the same as the visual ID column value for individual media packages. It will be blank for items.
PreviousVisualID	Varchar(40)	Y	Indicates the visual ID of the previous ticket/pass/or debit card when the package has been reprinted or upgraded. This is blank if the package detail was not created by reprinting or upgrading the package.
Deferred	Bit	Y	Indicates if the package detail was created as a deferred benefit when the package was sold.
Status <sup>3</sup>	Integer	Y	Status of the detail. Used only for item details to mark as voided or returned.
PrintSequence	Integer	Y	Sequence number to indicate the order in which the package details were printed. This allows package reprints to print detail information using the same keyword index that was used for the detail during the original sale.
NextVisualID	Varchar(40)	Y	Indicates the visual ID of the next ticket or pass when the package has been upgraded to something outside of a package. This allows tracking of lineage for usage purposes without affecting the Super Ticket chain.

#### Indexes

Name	Kind	Columns	Purpose
PKPkgInstDtlsPkgInstanceDetailID	P	PkgInstanceDetailID	Primary Key.
IXPkgInstDtlsVisualID	A	VisualID	To allow faster searching of a package instance detail for a given visual ID
IXPkgInstDtlsPiIDPkgDtlIDssSts	A	PkgInstanceID, PackageDetailID, IssuanceStatus	To allow faster searching for deferred details in a package where the PackageDetailID is known.
IXPkgInstDtlsTktTypeMediaVisID	A	TicketType, MediaVisualID	To speed up the query needed to load a debit card from a single media package.
IXPkgInstDtlsPreviousVisualID	A	PreviousVisualID	Allows faster searching of the next package instance detail in the lineage.
IXPkgInstDtlsNextVisualID	A	NextVisualID	Allows faster searching of the previous package detail in the lineage for a ticket or pass outside of a package.

#### <sup>1</sup> TicketType Values

Value	Gateway Constant Name	Description
0	ST_TICKET_TYPE	Ticket
1	ST_PASS_TYPE	Pass
3	ST_ITEM_TYPE	Item
4	ST_GTS_DEBIT_TYPE	GTS Debit
5	ST_DEBIT_TYPE	Debit (non-GTS debit)
6	ST_FEE_TYPE	Fee

#### <sup>2</sup> IssuanceStatus Values

Value	Gateway Constant Name	Description
0	ENTITLEMENT_STATUS_ISSUED	The entitlement was issued
1	ENTITLEMENT_STATUS_UNISSUED	The entitlement has not been issued

#### <sup>3</sup> Status Values

Value	Gateway Constant Name	Description
0	PKG_INSTANCE_DETAIL_VALID	Valid
1	PKG_INSTANCE_DETAIL_VOIDED	Voided
2	PKG_INSTANCE_DETAIL_RETURNED	Returned
3	PKG_INSTANCE_DETAIL_EXPIRED	Expired (For pass benefits packages)

### 3.131 PlanConversions

#### Columns

Column	Type	Allow Nulls	Description
PlanConversionID	Integer	N	Unique ID of the table.
PaymentPlanID	Integer	Y	ID of the original payment plan. Links to the PaymentPlans table.
ConvertPlanID	Integer	Y	ID of the payment plan that can be converted to. Links to the PaymentPlans table.

#### Indexes

Name	Kind	Columns	Purpose
PKPlanConversionsPlanConvrnID	P	PlanConversionID	Primary Key.
IXPlanConversionsPaymentPlanID	IX	PaymentPlanID	Index

### 3.132 PlanFops

#### Columns

Column	Type	Allow Nulls	Description
PlanFOPID	Integer	N	Primary Key, unique number.
PaymentPlanID	Integer	Y	ID of Payment Plan.
FOPCode	Integer	Y	FOP Code of the FOP that can be used with the Payment Plan.
PlanFOPGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPlanFopsPlanFOPID	P	PlanFOPID	Primary Key.

### 3.133 PlannerFilters

The PlannerFilters table contains all of the filters that have been defined for use within Planner. These filters are used to restrict the display to a subset of the available resource management information.

#### Columns

Column	Type	Allow Nulls	Description
PlannerFilterID	Int	N	Primary key, always unique
FilterName	VarChar(256)	Y	Name of the Planner Filter
AgencyNo	Int	Y	Agency number
UserID	Int	Y	ID of the user who created the filter; Foreign key to GxUsers.UserID
LineNum	Int	Y	Sequence value that defines the order in which the records should be used/displayed
ResourceTypeID	Int	Y	Resource Type referenced by the filter; Foreign key to RMResourceTypes.ResourceTypeID
ResourceId	Int	Y	Resource referenced by the filter; Foreign key to RMResources.ResourceID

#### Indexes

Name	Kind	Columns	Purpose
PKPlannerFilterID	P	PlannerFilterID	Primary Key
IXPlannerFiltersFilterName	IX	FilterName	
IXPlannerFiltersAgencyUserFilter	IX	AgencyNo, UserID, FilterName	

### 3.134 Plugin Attributes

The Plugin Attributes table connects attribute definitions to plugins.

#### Columns

Column	Type	Allow Nulls	Description
PluginAttributeID	Integer	N	Primary key, always unique
PluginID	Integer	N	Unique ID of the Plugin
AttributeDefinitionID	Integer	N	Unique ID of the Attribute Definition
AttributeGroupID	Int	N	Unique ID of the Attribute Definition Group

#### Indexes

Name	Kind	Columns	Purpose
PKPluginAttributeID	P	<i>PluginAttributeID</i>	Primary key - Unique ID in the table

### 3.135 PluginConfigSections

The PluginConfigSections table contains configuration information from plugins

#### Columns

Column	Type	Allow Nulls	Description
PluginConfigSectionID	Integer	N	Primary key, always unique.
PluginID	Integer	N	Foreign key to Plugins.PluginID. The unique ID of the plugin that the configuration data is associated to.
SectionID	Integer	N	ID defined by the plugin to separate configuration data.
SectionText	Text	Y	Block of configuration text from the plugin. The format is defined by the plugin. This value is encrypted if there is a key defined in the system. The key used is defined in the GxKeyID column.
GxKeyID	Integer	Y	Foreign key to GxMasterKeys.GxKeyID. Defines encryption scheme used to encrypt values for this record. This column will be zero if no columns are encrypted.

#### Indexes

Name	Kind	Columns	Purpose
PKPluginConfigSectionID	P	PluginConfigSectionID	Primary key - Unique ID in the table.
IXPCSPuginIDSectionID	IX	PluginID, SectionID	Alternate key. Unique.

### 3.136 Plugins

The Plugins table contains information about COM dlls that can be plugged into Galaxy to extend functionality.

#### Columns

Column	Type	Allow Nulls	Description
PluginID	Integer	N	Primary key, always unique.
RegisteredName	VarChar(100)	N	Name associated with the registered dll in the system registry.
PluginType	Integer	N	The type of functionality implemented by the plugin dll. <sup>1</sup>
Description	VarChar(30)	N	Description of the plugin. This will be used in Galaxy when referring to this plugin. For example, if the plugin is for a payment processor, this will be used as the display name for the protocol in Galaxy.
PluginGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPluginID	P	<i>PluginID</i>	Primary key - Unique ID in the table.

#### <sup>1</sup> PluginType Values

Value	Gateway Constant Name	Description
0	plugPaymentProcessor	The plugin is a payment processor implementing the IPaymentProcessor interface defined in Galaxy. This plugin will be implemented as a payment protocol in Galaxy.
1	plugLoyalty	Plugin for connecting Galaxy to an external loyalty system.
2	plugAdmissionControl	Plugin for external admission control.
3	plugXMLPaymentProcessor	The plugin is an XML payment processor implementing the IXMLPaymentProcessor interface defined in GTSXMLPaymentPlugin.dll. This is different from the old payment processor plugin because it uses a strictly XML interface that can be expanded without affecting compatibility of existing plugins. This plugin is implemented as a payment protocol in Galaxy.
4	plugIDEncoder	A plugin used to encode and decode visual IDs, order IDs, etc.
5	plugPostPaymentsProcessor	Plugin used for processing after payments.
6	plugPricing	Plugin used to determine the price of items.
7	plugEventLogExport	Plugin for external handling of event logs.
8	plugJournalExport	Plugin for external handling of journal records.
9	plugShift	Plugin for external handling of shift events.
10	plugReceipt	Plugin for external handling of receipt events.

### 3.137 POSCreditTransactions

This table contains records of the POS and Reseller transactions that used a Customer's Charge Account for payment.

#### Columns

Column	Type	Allow Nulls	Description
POSCreditTransactionID	Int	N	Primary Key.
JnlTranID	Int	N	Foreign key to JnlHeaders table
InvoiceID	Int	Y	Foreign key to ARInvoices table - is only populated if invoices are generated from within Galaxy
CreditAmount	Money	Y	The amount of credit used in the transaction
AccountID	Int	Y	Foreign key to ARAccounts table
CustomerID	Int	Y	Foreign key to Customers Table
CreditMemoid	Int	Y	Foreign key to ARCreditMemos table - is only populated if credit memos are generated from within Galaxy

#### Indexes

Name	Kind	Columns	Purpose
PK POSCreditTransactionID	P	POSCreditTransactionID	Primary Key.
IXPOSCrdtTransInvIDJnlTranID	IX	InvoiceID, JnlTranID	For queries by InvoiceID, JnlTranID

### 3.138 PostalCodeRangeDetails

This table stores a collection of postal code ranges defined by FromPostalCode and ThruPostalCode. Entering the same value in FromPostalCode and ThruPostalCode signifies a range of one postal code.

#### Columns

Column	Type	Allow Nulls	Description
PostalCodeRangeDetailID	Int	N	Primary Key.
PostalCodeRangeID	Int	N	Reference back to the <i>PostalCodeRanges</i> table.
FromPostalCode	VarChar(16)	Y	The start of the Postal Code range.
ThruPostalCode	VarChar(16)	Y	The end of the Postal Code range.
PostalCodeRangeDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPostalCodeRangeDetailID	P	PostalCodeRangeDetailID	Primary Key.

### 3.139 PostalCodeRanges

This table stores a header record for PostalCodeRangeDetails to enable grouping a series of postal code ranges together.

#### Columns

Column	Type	Allow Nulls	Description
PostalCodeRangeID	Int	N	Primary Key.
Name	VarChar(30)	Y	The meaningful name of the Postal code grouping.
Description	VarChar(256)	Y	More information on the collection of Post code ranges.
PostalCodeRangeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPostalCodeRangeID	P	PostalCodeRangeID	Primary Key.

### 3.140 PriceCalendars

This table is used to link a date/time to a price program.

#### Columns

Column	Type	Allow Nulls	Description
PriceCalendarUniqueID	Int	N	Primary key, always unique
PriceCalendarID	Int	N	
PriceProgramID	Int	N	FK reference to PricePrograms.PriceProgramID column
EffectiveDateTime	DateTime	N	Effective Date and Time of the PriceProgram
PriceProgramGroupID	Integer	N	Foreign key to PriceProgramGroups.PriceProgramGroupID. Indicates the price calendar is associated with a price program group. Zero indicates no price program group is associated.
PriceCalendarGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPriceCalendarUniqueID	P	PriceCalendarUniqueID	Primary Key.
IXPriceProgEffectiveDateTime	U	PriceProgramID, EffectiveDateTime	Unique key
IXPriceCalEffectiveDateTime		EffectiveDateTime	Used by the query to get Price program for a given date and time

### 3.141 PriceProgramGroups

This table defines a price program group.

#### Columns

Column	Type	Allow Nulls	Description
PriceProgramGroupID	Int	N	Primary key, always unique
Name	Varchar(80)	N	Name of the price program group
Inactive	Bit	N	True if price program group is Inactive, and not visible in most picklists.
AttributeValueGroupID	Integer	Y	Links price program groups to the AttributeValues table
PriceProgramGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPriceProgramGroupID	P	PriceProgramGroupID	Primary Key.

### 3.142 PricePrograms

This table defines a price program.

#### Columns

Column	Type	Allow Nulls	Description
PriceProgramUniqueID	Int	N	Primary key, always unique
Name	Varchar(80)	N	Name of the price program
Color	Integer	N	Color used to identify this Price Program on GUI screens
PriceProgramGroupID	Integer	N	Foreign key to PriceProgramGroups.PriceProgramGroupID. Indicates the price program group that is associated with the price program. Zero indicates no price program group is associated.
AttributeValueGroupID	Integer	Y	Links price programs to the AttributeValues table
Inactive	Bit	Y	True if Price Program is Inactive, and not visible in most picklists.
PriceProgramGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPriceProgramUniqueID	P	PriceProgramUniqueID	Primary Key.

### 3.143 PriceProgramTimeRanges

Defines a start and end time for a price program.

#### Columns

Column	Type	Allow Nulls	Description
PriceProgramTimeRangeUniqueID	Int	N	Primary key, always unique
PriceProgramTimeRangeID	Int	N	
Name	Varchar(80)	N	Name for the time range
StartTime	DateTime	N	Time range start time
EndTime	DateTime	N	Time range end time
PriceProgramTimeRangeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPriceProgramTimeRangeUnqID	P	PriceProgramTimeRangeUniqueID	Primary Key.

### 3.144 PriceSchedules

The PriceSchedules table defines price schedules based on PLU, ProcessCode, Price Program and time range.

#### Columns

Column	Type	Allow Nulls	Description
PriceScheduleUniqueID	Int	N	Primary key, always unique
PriceScheduleID	Int	N	
PriceProgramID	Int	N	FK reference to PricePrograms.PriceProgramID
PriceProgramTimeRangeID	Int	N	FK reference to PriceProgramTimeRanges.PriceProgramTimeRangeID column
DisbursementID	Int	Y	FK reference to Disbursements.DisbursementID column
Price	Money	N	Price to use for the given PLU
ProcessCode	Int	N	Process that will use this Price Schedule <sup>1</sup>
PLU	Char (20)	N	PLU of the item
ReplacementAccessCode	Int	Y	When an item is sold using the Price Program/Price Program Time Range for this record, its Access Code will be replaced with this one.
ReplacementCompany	Int	Y	When an item is sold using the Price Program/Price Program Time Range for this record, its company will be replaced with this one.
ReplacementCategory	Int	Y	When an item is sold using the Price Program/Price Program Time Range for this record, its category will be replaced with this one.
ReplacementSubCategory	Int	Y	When an item is sold using the Price Program/Price Program Time Range for this record, its subcategory will be replaced with this one.
PriceScheduleGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### <sup>1</sup> ProcessCode Values

Value	Gateway Constant Name	Description
0	TKT_SALE_PROCESS_CODE	Price Schedule is used when a ticket is sold (at POS, or OrderEntry)
1	TKT_UPGRADE_PROCESS_CODE	Price Schedule is used when a ticket is upgraded (Ticket Upgrade function)

#### Indexes

Name	Kind	Columns	Purpose
PKPriceScheduleUniqueID	P	PriceScheduleUniqueID	Primary Key.
AKPriceSchedUniqueKey	U	PLU, ProcessCode, PriceProgramID, PriceProgramTimeRangeID	Unique key.
IXPriceSchedPLUPCRangeIDPrice		PLU, ProcessCode, PriceProgramTimeRangeID, Price	Used by the query to get price for a PLU

### 3.145 ProfileControls

This table is used to store references to each profile that has "control" over another. Each row basically consists of two user profile ID numbers. The first (UserProfileID), is the user profile that "controls" another. The second, (ControlledUserProfileID) is the profile that is being "controlled".

Column	Type	Allow Nulls	Description
ProfileControlID	Int	N	Primary key, always unique. System generated.
UserProfileID	Int	N	Foreign key to UserProfiles.UserProfileID. User profile ID the is controlling.
ControlledUserProfileID	Int	N	Foreign key to UserProfiles.UserProfileID. User profile ID that is being controlled.
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKProfileControlsControlID	P	ProfileControlID	Primary key.
IXProfileControlsUserProfileID		UserProfileID	Currently not used by the system. Can be used to query on all the profiles a given profile "controls".

#### <sup>1</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore profile control records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store ProfileControls.

### 3.146 ProfilePrvs

This table is used to store the individual privileges that are enabled for each user profile.

Column	Type	Allow Nulls	Description
ProfilePrivID	Int	N	Primary key, always unique. System generated.
UserProfileID	Int	Y	Foreign key to UserProfiles.UserProfileID
PrivNumber	Int	Y	Galaxy privilege number. Each user privilege has a unique number.
PrivilegeText	VarChar(100)	Y	Galaxy profile text
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>1</sup>
ProfilePrivGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKProfilePrvsProfilePrivID	P	ProfilePrivID	Primary key.
IXProfilePrvsUserProfileID		UserProfileID	Currently not used by the system. Can be used to query on all the privileges for a given profile ID.

#### <sup>1</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprvs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore ProfilePriv records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store ProfilePrvs.

### 3.147 Prompts

This table used to store the Prompts defined in Galaxy.

Column	Type	Allow Nulls	Description
PromptUniqueID	Int	N	Primary key, always unique. System generated.
PromptID	Char(4)	Y	
Description	Char(30)	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKPromptsPromptUniqueID	P	PromptUniqueID	Primary key.

### 3.148 PromptSections

This table used to store prompt sections defined in Galaxy.

Column	Type	Allow Nulls	Description
PromptSectionID	Int	N	Primary key, always unique. System generated.
PromptID	Int	N	
SectionStart	Int	Y	
SectionLength	Int	Y	
SectionFormat	Int	Y	
SectionValue	Char(40)	Y	
Store	Bit	Y	
Display	Bit	Y	
EnterOnly	Bit	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKPromptSectionsPrmpSectionID	P	PromptSectionID	Primary key.

### 3.149 ProtocolIniSections

The table stores the payment protocol information from the Protocol.ini file. A single row contains the information in one section in the ini file.

#### Columns

Column	Type	Allow Nulls	Description
ProtocolIniSectionID	Int	N	Primary key, always unique
ProtocolName	NVarChar(10)	N	Name of the protocol.
SectionName	NVarChar(50)	N	Name of the section in the Protocol.ini file.
SectionText	Text	Y	Raw text from the Protocol.ini file section. This holds a comma-separated list of keys and values from the ini file section.

#### Indexes

Name	Kind	Columns	Purpose
PKProtocolIniSectionID	P	ProtocolIniSectionID	Primary Key, always unique
IXProtocolISProtNameSection	A	ProtocolName, SectionName	Alternate Key

### 3.150 Protocols

The table stores the payment protocol information.

#### Columns

Column	Type	Allow Nulls	Description
ProtocolID	Int	N	Primary key, always unique
Name	NVarChar(10)	N	Name of the protocol. This either comes from a fixed set of protocol names, or contains a plugin ID for plugin protocols.
Description	NVarChar(60)	Y	Protocol description.
MultiTrans	Bit	Y	Not used.
DataBits	Int	Y	The number of data bits for the protocol communications.
StopBits	Int	Y	The number of stop bits for the protocol communications.
Baud	Int	Y	The baud rate for the protocol communications.
Parity	Int	Y	The parity for the protocol communications. <sup>1</sup>
TimerConnect	Int	Y	The dialup connection timer (in seconds).
TimerHostENQ	Int	Y	The dialup host ENQ timer (in seconds).
TimerResponse	Int	Y	The dialup host response timer (in seconds).
TimerHostEOT	Int	Y	The dialup host EOT timer (in seconds).
RequestTries	Int	Y	The maximum number of request attempts.
ResponseTries	Int	Y	Not defined. Use differs depending on the protocol.
NoEOTDiscard	Bit	Y	Discard response on an EOT timeout.
DecimalAmount	Bit	Y	Not used.
NormalTimeout	Int	Y	Host response timeout setting.
CloseTimeout	Int	Y	Cashout response timeout setting.
FloorLimit	Money	Y	The floor limit amount for the protocol.
OfflineFloorLimit	Bit	Y	Indicates if the floor limit is for offline transactions.
CurrentConversion	Int	Y	Indicates the current conversion version for the protocol record.
HostURL	NVarChar(255)	Y	The address (URL) of the host payment server.
HostField1	NVarChar(255)	Y	Host data field 1.
ProtocolType	Int	Y	Indicates the protocol type <sup>2</sup>
HostField2	NVarChar(255)	Y	Host data field 2.
ConvertFOP	Bit	Y	Indicates if the FOP should be changed to match the card range for the account number returned from the host. Used for XML payment plugin protocols.
DebitFOPID	Int	Y	Indicates the FOP code for the FOP that should be used if the transaction at the host is a debit transaction. Used for XML payment plugin protocols.
SkipReturnPrompt	Bit	Y	Indicates if the auth code prompt should be skipped on a return. Used for XML payment plugin protocols.
HostTktSetID1	Int	Y	Foreign Key to TktSets.TktSetNo. ID of a ticket set used to generate a receipt coupon to send to a host during payment authorization. Used by payment plugins.
HostTktSetID2	Int	Y	Foreign Key to TktSets.TktSetNo. ID of a ticket set used to generate a second receipt coupon to send to a host during payment authorization. Used by payment plugins.

#### Indexes

Name	Kind	Columns	Purpose
PKProtocolsProtocolID	P	ProtocolID	Primary Key, always unique
IXProtocolsName	A	Name	Alternate Key

#### <sup>1</sup> Parity Values

Value	Gateway Constant Name	Description
-1	NO_PARITY	No parity
0	ODD_PARITY	Odd parity
1	EVEN_PARITY	Even parity

#### <sup>2</sup> ProtocolType Values

Value	Gateway Constant Name	Description
0	ptStandard	Standard internal protocol
1	ptPlugin	Protocol is implemented by a plugin dll
2	ptXMLPlugin	Protocol is implemented by an XML plugin dll

### 3.151 PublishStatus

This table is used to keep track of the number of times a user has published data. This value, in combination with the RowChanges table will provide a 'batch' of data changes. The table will currently have at most one row.

#### Columns

Column	Type	Allows Nulls	Description
PublishStatusID	Int	N	Primary key, always unique
PublishCount	Int	N	The number of times data has been published
LastPublish	Datetime	N	The date/time of the last publish

#### Indexes

Name	Kind	Columns	Purpose
PK_PublishStatusID	P	PublishStatusID	Primary Key.

### 3.152 Reasons

This table is used in the maintenance of Reasons.

#### Columns

Column	Type	Allow Nulls	Description
ReasonID	Integer	No	Primary key, always unique
ID	Integer	No	Local key
SurveyID	Integer	Yes	Optional survey ID
Reason	Varchar(20)	Yes	Short Reason text
Description	Varchar(60)	Yes	Long Reason text
MaxTickets	Integer	Yes	Populated when the number of tickets sold using this reason is limited to a specified quantity.
ReasonType	Integer	Yes	Allows for different categories of reasons <sup>1</sup>
Inactive	Bit	Yes	Indicates if the reason is currently inactive.

#### Indexes

Name	Kind	Columns	Purpose
PKReasonID	P	ReasonID	Primary key, must be unique.

<sup>1</sup> ReasonType Values

Gateway Constant Name	Value	Description
rtTicketReason	0	Ticket sale reason
rtPaymentRefundReason	1	Reason for payment refund

### 3.153 RecurrencePatterns

Contains the recurrence pattern data for Payment Contracts.

#### Columns

Column	Type	Allow Nulls	Description
RecurrencePatternID	Integer	N	Primary Key, unique number.
RecurrenceID	Integer	Y	ID used by Btree (Not used currently).
Name	VarChar(50)	Y	Recurrence Name
DayOfMonth	Integer	Y	Day of Month selected. Can be any integer between 1 and 31.
DayOfMonthMask	Integer	Y	Selected days of the month (Not used currently).
DayOfWeekMask	Integer	Y	Stores day(s) of the week (Sunday thru Saturday) (Not used currently).
Instance	Integer	Y	Will be used by Month and Year (Not currently used).
Interval	Integer	Y	(n) number of RecurrenceTypes. If RecurrenceType is Daily, this is the number of days. If Monthly, this is the number of months. Currently, can only be 1 for Monthly.
MonthOfYear	Integer	Y	Selected month of the year (Not currently used).
RecurrenceType	Integer	Y	Indicates the frequency of payments. <sup>1</sup>
StartDate	DateTime	Y	Start date of recurrence.
EndDate	DateTime	Y	Ending date of recurrence.
EnableTime	Bit	Y	Turn on time checking of recurrence (Not currently used).
StartTime	DateTime	Y	Start time of recurrence if time is enabled.
EndTime	DateTime	Y	End time of recurrence if time is enabled (not currently used).
RecurrencePatternGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKRecurrencePatternsRecPattID	P	RecurrencePatternID	Primary Key.

#### <sup>1</sup> RecurrenceType Values

Value	Gateway Constant Name	Description
0	NONE_RECURRENCE_TYPE	No recurrence type
1	DAILY_RECURRENCE_TYPE	Payment Contract payments due daily.
2	WEEKLY_RECURRENCE_TYPE	Payment Contract payment due weekly.
3	WEEKLY_NTH_RECURRENCE_TYPE	Payment Contract payment due every n weeks.
4	MONTHLY_RECURRENCE_TYPE	Payment Contract payment due monthly.
5	MONTHLY_NTH_RECURRENCE_TYPE	Payment Contract payment due every n months.
6	YEARLY_RECURRENCE_TYPE	Payment Contract payment due annually.

### 3.154 Relationships

A relationship record links two contacts together via a relationship type.

#### Columns

Column	Type	Allow Nulls	Description
RelationshipID	Int	N	Primary key, always unique
RelationshipTypeID	Int	N	ID from RelationshipType table
FromContactID	Int	N	From contact id; link to CustContacts.CustContactID
FromType	Int	Y	Contact type - not used at this time
FromDate	DateTime	N	Start date of relationship
ToContactID	Int	N	To contact id; link to CustContacts.CustContactID
ToType	Int	Y	Contact type - not used at this time
ToDate	DateTime	N	End date of relationship

#### Indexes

Name	Kind	Columns	Purpose
PKRelationshipID	P	RelationshipID	Primary Key.
IXRelationshipsRelationshipTypeID		RelationshipTypeID	Loadup all relationships via Relationship Type
IXRelationshipsFromContactID	A	FromContactID	Index used for loading relationships by FromContactID
IXRelationshipsToContactID	A	ToContactID	Index used for loading relationships by ToContactID

### 3.155 RelationshipTypes

Defines how a contact is related to another contact.

#### Columns

Column	Type	Allow Nulls	Description
RelationshipTypeID	Int	N	Primary key, always unique
Description	Char(50)	Y	Description of relationship type
ReciprocalID	Int	N	Relationship Type id of reciprocal
Sequence	Int	Y	This field is used to store an integer value which represents display sequence when displaying relationship types on Edit Pass form.
Inactive	Bit	Y	This field is the Attribute Inactive. 0 = Active, 1 = Inactive
RelationshipTypeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKRelationshipTypeID	P	RelationshipTypeID	Primary Key.

### 3.156 RenewalOptions

This table contains the Renewal items that the current item can renew to. This ties a Pass or Pass Renewal item to Pass Renewal items that are valid to renew to.

#### Columns

Column	Type	Allow Nulls	Description
RenewalOptionID	Int	N	Primary key, always unique
PLUSource	Char(20)	N	PLU that is the starting point for the renewal transaction
PLURenewal	Char(20)	N	PLU to renew to if this renewal option is selected
UseForPaymentContract	Bit	Y	Indicates that the PLU in the PLURenewal column is to be used for Payment Contract renewals
RenewalEffectiveDate	DateTime	Y	Renewal effective date
RenewalExpirationDate	DateTime	Y	Renewal expiration date
ItemGroupID	Int	Y	Foreign key link to ItemGroups.ItemGroupID
Sequence	Int	Y	Allows users to determine presentation order in PassPortal operations.
Availability <sup>1</sup>	Int	Y	Allows specifying when the renewal option is available relative to the expiration date
RenewalOptionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKRenewalOptionID	P	RenewalOptionID	Primary Key.

#### <sup>1</sup> Availability Values

Value	Gateway Constant Name	Description
0	RENEWAL_OPTION_AVAILABLE_ALWAYS	Always Available
1	RENEWAL_OPTION_AVAILABLE_WITHIN_WINDOW	Within the Renewal Window Only
2	RENEWAL_OPTION_AVAILABLE_AFTER_WINDOW	After the Renewal Window Only
3	RENEWAL_OPTION_AVAILABLE_WITHIN_EXPIRATION_WINDOW	Within the Expiration Window Only
4	RENEWAL_OPTION_AVAILABLE_AFTER_WINDOW_END	After the Expiration Window Only

### 3.157 ReportParameterDetails

This table holds the choices for the parameter if it has multiple defined choices.

#### Columns

Column	Type	Allow Nulls	Description
ReportParameterDetailID	Int	N	Unique Identifier
ReportParameterID	Int	N	Link to ReportParameters table. There can be 1 to many entries in this table for a given ReportParameterID
Answer	VarChar(255)	N	The answer for this multiple choice entry
DefaultAnswer	Bit	Y	Is this the default selection

#### Indexes

Name	Kind	Columns	Purpose
PK_ReportParameterDetailID	P	ReportParameterDetailID	

### 3.158 ReportParameters

This table holds the parameters required to create the report

#### Columns

Column	Type	Allow Nulls	Description
ReportParameterID	Int	N	Unique Identifier
ReportID	Int	N	Foreign Key to Reports table
Name	VarChar(100)	N	The name of the parameter
Description	VarChar(255)	Y	The description of the parameter
DataType	Int	N	The type of data the parameter is. This will be an integer which defines the type of the parameter. Example: Integer, String, Multiple Choice, etc.
Size	Int	Y	Defines the size of the parameter. Only really applicable if Type is a String.
LinkedTable	VarChar(100)	Y	If this parameter is a foreign key reference to another table, this column defines the name of the table.
LinkedFieldName	VarChar(100)	Y	If this parameter is a foreign key reference to another table, this column defines the name of the column it refers to.
Required	Bit	Y	Is this parameter required to be filled in?

#### Indexes

Name	Kind	Columns	Purpose
PKReportParameterID	P	ReportParameterID	

NOTE - When this table is created/updated values defining the reports are automatically inserted into SQL.

### 3.159 Reports

This table holds the list of available reports and their descriptions

#### Columns

Column	Type	Allow Nulls	Description
ReportID	Int	N	Unique Identifier
ReportName	VarChar(100)	N	The name of the report
Description	VarChar(255)	Y	Description for the report
ReportType <sup>1</sup>	Int	Y	Integer constant identifying the report

#### Indexes

Name	Kind	Columns	Purpose
PKReportsReportID	P	ReportID	

<sup>1</sup> ReportType Values

Value	Gateway Constant Name	Description
1	REPORT_TYPE_RESELLER_SALES_SUMMARY	Reseller sales summary report
2	REPORT_TYPE_RESELLER_SALES_DETAIL	Reseller sales detail report
3	REPORT_TYPE_RESELLER_RETURN_DETAIL	Reseller return detail report
4	REPORT_TYPE_RESELLER_USAGE	Reseller usage report
5	REPORT_TYPE_RESELLER_CONSOLIDATED_SALES_SUMMARY	Reseller consolidated sales summary report
6	REPORT_TYPE_RESELLER_CONSOLIDATED_CATEGORY_GROUP	Reseller consolidated category group report

NOTE - When this table is created/updated values defining the reports are automatically inserted into SQL.

**3.160 ReservationStatus****Columns**

Column	Type	Allow Nulls	Description
ResStatusID	Int	N	
FacilityID	Int	Y	Foreign key to Facility.IdNo
ReservationDate	DateTime	Y	
MaxReservations	Int	Y	
UsedReservations	Int	N	

**Indexes**

(none)

### 3.161 Rosters

Used to collect values for the roster attributes associated with events and contacts.

#### Columns

Column	Type	Allow Nulls	Description
RosterID	Int	N	Primary key, always unique
EventID	Int	N	Reference to associated Event
ContactID	Int	N	Reference to CustContacts.CustContactID
RosterAttributeValueGroupID	Int	N	Reference to associated Attribute Values
OrderLineID	Int	Y	Foreign key to OrderLines.OrderLineID. Links the roster record to an order line.

#### Indexes

Name	Kind	Columns	Purpose
PKRostersRosterID	P	ShiftStatusID	Primary Key.
IXRostersContactID		ContactID	Foreign key to CustContacts.CustContactID
IXRostersOrderLineID		OrderLineID	Foreign key to OrderLines.OrderLineID

### 3.162 RosterTemplates

A table that will provide a template for rosters, most notably attributes.

#### Columns

Column	Type	Allow Nulls	Description
RosterTemplateID	Int	N	Primary key, always unique
Name	Varchar(30)	N	Name of Roster Template
RosterAttributeGroupID	Int	N	Reference to AttributeGroups table

#### Indexes

Name	Kind	Columns	Purpose
PKRstrTmpltRstrTmpltID	P	RosterTemplateID	Primary Key.

### 3.163 RowChanges

This table is used to store indicators for each row of configuration data that is changed. When a row in a configuration table is inserted, updated or deleted, a row is created in this table, pointing to that changed row.

#### Columns

Column	Type	Allow Nulls	Description
RowChangeID	Int	N	Primary key, always unique
RowID	Int	N	The row that was changed in the table specified by TableID
TableID	Int	N	A specific table that changed. The row in this table is specified by TableID.
Code	Int	N	This value corresponds to ConfigurationOptions.Code. It allows for easy joins when looking for changes for tables that are centrally managed
PublishCount	Int	N	The corresponding publish ID in which this change occurred
Action <sup>1</sup>	Int	N	The row was inserted, deleted, updated.
LocalKey	Varchar(30)	N	The unique identifier for the row in the local data

#### Indexes

Name	Kind	Columns	Purpose
PK RowChangeID	P	RowChangeID	Primary Key.
IXRowChangesLastUpdate	IX	LastUpdate	To improve performance on queries to truncate this table by date range.
IXRowChangesCode	IX	Code	Improve query performance

#### <sup>1</sup> Action Values

Value	Description
1	A row was inserted into the table represented by TableID.
2	A row was deleted from the table represented by TableID.
3	A row was updated in the table represented by TableID.

### 3.164 SaleLimits

The SaleLimits table contains the header information for grouping sale limits. The details for sale limits are stored in the SaleLimitDetails table.

#### Columns

Column	Type	Allow Nulls	Description
SaleLimitID	Int	N	Primary key, always unique.
Name	VarChar(50)	N	Name of the sale limit.

#### Indexes

Name	Kind	Columns	Purpose
PKSaleLimitID	P	SaleLimitID	Primary Key.

### 3.165 SaleLimitDetails

The SaleLimitDetails table contains the detail information for sale limits. The header information is stored in the SaleLimits table.

#### Columns

Column	Type	Allow Nulls	Description
SaleLimitDetailID	Int	N	Primary key, always unique.
SaleLimitID	Int	N	Foreign key to the SaleLimits.SaleLimitID.
LimitType	Int	N	The type of sale limit. <sup>1</sup>
Limit	Int	N	The quantity allowed for this sale limit type.

#### Indexes

Name	Kind	Columns	Purpose
PKSaleLimitDetailID	P	SaleLimitDetailID	Primary Key.
IXSaleLimitDetailsSaleLimitID		SaleLimitID	Used to find all detail records for a given SaleLimitID.

<sup>1</sup> LimitType Values

Value	Gateway Constant Name	Description
0	ltDaily	Daily limit.
1	ltWeekly	Weekly limit.
3	ltMonthly	Monthly limit.

### 3.166 SettlementBatchDetails

The **SettlementBatchDetails** table is used to store detail information related to settlement batches created for the VisaNet authorization protocol. Generally this table will contain one row per transaction with an authorized credit card. Transactions that have multiple authorizations will have multiple rows in this table.

#### Columns

Column	Type	Allow Nulls	Description
SettlementBatchDetailID	Int	N	Primary key, always unique
SettlementBatchID	Int	N	FK into the SettlementBatches table
JnlTranID	Int	N	Reference to the transaction that contained this authorization. References JnlHeaders.JnlTranID (in addition to many other journal tables with a JnlTranID column)
NodeNo	Int	N	POS node number where this transaction was recorded
TranNo	Int	N	POS transaction number for this authorization
FiscalDate	Datetime	N	The date/time recorded by the POS for this authorization
Amount	Money	N	The amount authorized
AccountSource	Int	Y	Indicates how the card number was captured by the system (swiped in an MSR, manual, etc). See JnlDrafts.AccountSource for possible values.
AuthCode	Varchar(128)	Y	The authorization code recorded by the POS upon a successful authorization
CardNo	Varchar(255)	Y	The encrypted credit card number
TransID	Varchar(255)	Y	The host-specific transaction number returned in the authorization response
Validation	Varchar(255)	Y	Host specific field returned in the authorization response
SettlementCode	Varchar(255)	Y	Host specific data returned in the authorization response that will be included in the settlement message to the host
SettlementCode2	Varchar(255)	Y	Additional host specific data returned in the authorization response that will be included in the settlement message to the host
FOP	Int	Y	The form of payment number used to tender the payment at the POS
Captured	Bit	Y	True if the value is captured by the host - this will generally be false for protocols that are terminal capture. See JnlDrafts.Capture for more information
GxKeyID	Int	Y	The ID of the encryption key used to encrypt the credit card number
Status	Int	Y	A status indicator for the batch detail <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKSettlementBatchDetailID	P	SettlementBatchDetailID	Primary key.
IXSBDJnlTranIDAuthCode	IX	JnlTranID, AuthCode	To speed up queries for the combination of JnlTranID and AuthCode which uniquely identify a payment.

#### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	SETTLEMENT_DETAIL_INCLUDED_STATUS	The detail (authorization) will be included when the batch is settled
1	SETTLEMENT_DETAIL_EXCLUDED_STATUS	The detail will be excluded from the batch when settled - this value would be used to remove an authorization from a batch
2	SETTLEMENT_DETAIL_VOIDED_STATUS	The detail was added to a batch, but later was voided on a POS - it will not be included in the settlement process

### 3.167 SettlementBatches

The **Settlement Batches** table is used to store header information related to settlement batches created for the VisaNet authorization protocol. One row per settlement batch will be created in this table.

#### Columns

Column	Type	Allow Nulls	Description
SettlementBatchID	Int	N	Primary key, always unique
BatchNo	Int	N	The number assigned to the batch. It will be an incremental number that rolls over at 999. The counter VisaNetBatchNo from GatewayCounters is used to keep track of the next number.
CreateDate	Datetime	N	The date the batch was created
BatchStatus	Int	Y	The status of the batch. See below for possible values <sup>1</sup>
BatchCount	Int	Y	The number of details in the batch, generally, this will be the number of transactions in the batch
Response	Varchar(50)	Y	This column will contain the response code received from the host after a settlement
ResponseMessage	Varchar(255)	Y	This column will contain the response message received from the host after a settlement
SettlementDate	Datetime	Y	The date that the batch was settled
SettlementStatus	Int	Y	The result of the settlement. See below for possible values <sup>2</sup>
SettlementAmount	Money	Y	The amount in the batch that was settled. This will only be populated after the batch was settled successfully, otherwise it will be zero
SettlementCount	Int	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKSettlementBatchID	P	SettlementBatchID	Primary key.

#### <sup>1</sup> BatchStatus Values

Value	Gateway Constant Name	Description
0	BATCH_PENDING_STATUS	Settlement on this batch is pending, meaning it has been created, but not transmitted for settlement yet
1	BATCH_POSTED_STATUS	This batch has been transmitted for settlement. The SettlementStatus value will indicate the result of the settlement

#### <sup>2</sup> SettlementStatus Values

Value	Gateway Constant Name	Description
0	SETTLEMENT_NOTSETTLED_STATUS	Indicates that the batch has not been settled
1	SETTLEMENT_SUCCESS_STATUS	The batch was settled successfully
2	SETTLEMENT_FAIL_STATUS	The settlement for this batch was attempted and failed. Response and response text will contain information regarding the failure
3	SETTLEMENT_RETRY_STATUS	The submitted batch came back with a status of referral
4	SETTLEMENT_DUP_STATUS	Batch was already settled, another attempt was made and the host returned a duplicate response

### 3.168 SettlementConfig

For supported protocols, The **SettlementConfig** table contains all protocol-specific SQL settings that are global to all stations, and/or specific to the settlement process. Depending on the protocol, some of this data may be copied to local Terminal ID records during Galaxy's Terminal ID download process.

#### Columns

Column	Type	Allow Nulls	Description
SettlementConfigID	Int	N	Primary key, always unique
Protocol	Char(10)	N	The name of the protocol <sup>1</sup>
Name	VarChar(20)	N	Name of configuration option <sup>2</sup>
Value	VarChar(128)	N	Value of configuration option

#### Indexes

Name	Kind	Columns	Purpose
PK SettlementConfigID	P	SettlementConfigID	Primary key.

#### <sup>1</sup> Protocol Values

Value	Gateway Constant Name	Description
HYPERCOM	HYPERCOM_SQL_PROTOCOL	Corresponds to the HYPERCOM1 and HYPERCOM2 protocols (Hypercom ISO 8583 authorization)

#### <sup>2</sup> Name Values

Protocol	Name Value	Gateway Constant Name	Description
HYPERCOM	Address1	HYPERCOM_ADDRESS_1	The address of the primary (HYPERCOM1) host.
HYPERCOM	PortNumber1	HYPERCOM_PORT_NUMBER_1	Port number of the primary (HYPERCOM1) host.
HYPERCOM	Address2	HYPERCOM_ADDRESS_2	The address of the secondary (HYPERCOM2) host.
HYPERCOM	PortNumber2	HYPERCOM_PORT_NUMBER_2	Port number of the secondary (HYPERCOM2) host.
HYPERCOM	BatchSizeLimit	HYPERCOM_BATCH_SIZE_LIMIT	The maximum size of a daily batch for a node.
HYPERCOM	BatchRetries	HYPERCOM_BATCH_RETRIES	Number of times to retry a failed settlement attempt.
HYPERCOM	NotifyEmail	HYPERCOM_NOTIFY_EMAIL	Email address to send Hypercom-related errors.

### 3.169 Signatures

The Signatures table contains data for signatures obtained from a signature capture device during a Galaxy Point of Sale transaction.

#### Columns

Column	Type	Allow Nulls	Description
SignatureID	Int	N	Primary key, always unique
TransactionNumber	Int	Y	Transaction number that the signature was captured in
NodeNo	Int	Y	The node that the signature was captured at
TransactionDate	DateTime	Y	The date/time of the transaction
FOP	Int	Y	Form of payment number that the signature was captured for
FOPSequence	Int	Y	Used to uniquely identify a payment. Will be 1 for single payment transactions. For multi-payment transactions will be the sequential number of the payment within the transaction.
DataFormat <sup>1</sup>	Int	Y	The format of SignatureData. See existing formats below
SignatureData	Int	Y	The actual signature stored in the format indicated by DataFormat

#### Indexes

Name	Kind	Columns	Purpose
PKSignaturesSignatureID	P	SignatureID	Primary Key.
IXSigTransNoNodeNo	IX	TransactionNumber, NodeNo	Used to query for payments per transaction. Typically, there will be one authorized payment per transaction. In the rare situation at there is multiple, only a few records (2 or 3) will exist for the given transaction and node.

<sup>1</sup> DataFormat Values

Value	Description
1	PointArray
2	JPEG
3	Bitmap (1 bit)

### 3.170 SiteItems

The SiteItems table associates item information (PLU or ItemGroupID) with a particular site. This table is used by the Export Service Module.

#### Columns

Column	Type	Allow Nulls	Description
SiteItemID	Int	N	Primary key, always unique
SiteUniqueID	Int	N	Reference to Sites.SiteUniqueID
PLU	Char(20)	Y	Item PLU
ItemGroupID	Int	Y	Reference to GxItemGroupDetails.ItemGroupID

#### Indexes

Name	Kind	Columns	Purpose
PKSiteItemsSiteItemID	P	SiteUniqueID	Primary Key

### 3.171 SiteMessages

The SiteMessages table associates Messages (e.g., ImportTicket, TicketUsage) with a particular site.

#### Columns

Column	Type	Allow Nulls	Description
SiteMessageID	Int	N	Primary key, always unique
SiteUniqueID	Int	N	Reference to Sites.SiteUniqueID
ExternalConnectionID	Int	N	Reference to ExternalConnections. ExternalConnectionID
ExternalSourceID	VarChar(128)	N	Matches SourceID field in message header. Must contain a unique value for each source.
MessageType <sup>1</sup>	Int	N	Indicates the message type (see values) <sup>1</sup>
MessageXSLT	VarChar(8000)	Y	Contents of the XSLT document to transform the message
FTPTargetDir	VarChar(256)	Y	FTP Directory where files will be uploaded
FTPNameFormat	VarChar(128)	Y	Keyword based string defining uploaded FTP file name format (including extension)
ScheduleTime	Int	Y	The time of day to process SiteMessages, formatted as hhmm, or zero if the processing will be performed in real-time according to the configured service interval.
ContentType	VarChar(255)	Y	Custom ContentType, used for messages sent via HTTP

#### Indexes

Name	Kind	Columns	Purpose
PKSiteMessagesSiteMessageID	P	SiteUniqueID	Primary Key

#### <sup>1</sup> MessageType Values

Value	Gateway Constant Name	Description
1	SITEMESSAGE_TYPE_TICKET	Ticket Creation Message
2	SITEMESSAGE_TYPE_USAGE	Usage Report Message

### 3.172 Sites

The sites table contains information about all the sites the current attraction is associated with, and is used by the Galaxy MultiSite and Export Service modules to synch sales and usage data.

#### Columns

Column	Type	Allow Nulls	Description
SiteUniqueID	Int	N	Primary key, always unique
SiteID	Varchar(30)	N	SiteID of an attraction as defined by the Gateway customer in the MultiSite module.
Name	VarChar(256)	N	Name of the site or attraction
GalaxySiteID	Int	N	SiteID of an attraction as defined by Gateway Ticketing Systems Inc.

#### Indexes

Name	Kind	Columns	Purpose
PKSitesSiteUniqueID	P	SiteUniqueID	Primary Key
AKSitesGalaxySiteID	A	GalaxySiteID	Unique key

### 3.173 Stocks

The ticket **stock** database contains the ticket stock types that is associated with a ticket-type to allow tracking of the preprinted tickets.

#### Columns

Column	Type	Allow Nulls	Description
StockUniqueId	Int	N	Primary key, always unique. System generated.
StockNo	Int	N	User definable ticket stock number - must be unique.
StockDescr	Char(30)	Y	Description of stock.
StockQty	Int	N	Number of coupons in pack.
Nkinds	Int	N	Number of stock kinds in package.
Preprinted	bit	N	If TRUE, this ticket stock is pre-printed.

#### Indexes

Name	Kind	Columns	Purpose
PKStocksStockUniqueId	P	StockUniqueId	Primary key
IXStocksStockNo		StockNo	Unique index on StockNo, as this field was unique in BTree

### 3.174 StockPackages

The StockPackages stores detail type information for ticket stock records

#### Columns

Column	Type	Allow Nulls	Description
StockPackageID	Int	N	Primary key, always unique. System generated.
StockUniqueID	Int	Y	The ID number that can be used to find all the package detail records for a given stock record. FK to Stocks.StockUniqueID.
PackageStockNo	Int	Y	The stock number for this package entry. Refers to Stocks.StockNo.
Qty	Int	Y	Quantity of the stock package

#### Indexes

Name	Kind	Columns	Purpose
PKStockPackagesStockPackageID	P	StockPackageID	Primary key

### 3.175 SurveyFields

This table contains the fields for a survey.

#### Columns

Column	Type	Allow Nulls	Description
SurveyFieldID	Int	N	Primary key, always unique
SurveyID	Int	N	ID of the survey associated with this record
FieldLabel	VarChar(240)	Y	Label for the survey field
FieldType	Int	Y	Survey field Type <sup>1</sup>
MultipleChoiceID	Int	Y	Multiple Choice for this field, FK to MultipleChoices.MultipleChoiceID
FieldHelp	Char(74)	Y	Help text
Required	Bit	Y	Determines if the field is required or not
FieldWidth	Int	Y	Maximum width of the field
FieldWidth2	Int	Y	Maximum number of decimal places, used when FieldType is 2
Lockout	Bit	Y	Determines if the entered ticket VisualID is added to the Lockout table, only used when FieldType is 5
ReferenceStart	Int	Y	Starting position of the reference field within the answer text
ReferenceLength	Int	Y	Length of the reference field within the answer text
PageNumber	Int	Y	Assigns a survey question to a specific page number
FieldWeight	Int	Y	The weight that this particular question carries in relation to the other questions in the survey. The total weight of all the questions in a survey must add up to be 100%.
Sequence	Int	Y	The sequence number of the field. This is stored to maintain the order of the questions in the survey.
SurveyFieldGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSurveyFieldsSurveyFieldID	P	SurveyFieldID	Primary key.

#### <sup>1</sup> FieldType Values

Value	Gateway Constant Name	Description
0	UNUSED_KIND	Field not used
1	STRING_KIND	Survey field is a string
2	NUMERIC_KIND	Survey field is a number
3	LOGICAL_KIND	Survey field is a boolean
4	MCHOICE_KIND	Survey field is a multiple choice
5	TICKET_KIND	Survey field is a ticket

### 3.176 SurveyReferences

This table contains References built from survey answers

#### Columns

Column	Type	Allow Nulls	Description
SurveyReferenceID	Int	N	Primary key, always unique.
SurveyID	Int	N	ID of the survey associated with this record
Reference	Char(255)	N	Reference Key built from survey answers.
NodeNo	Int	N	Node Number from where the record originated.
TranNo	Int	N	Transaction ID of the Journal transaction.
ReasonID	Int	N	ID of the reason associated with this record
OverridelD	Int	N	ID of the override associated with this record
Qty	Int	Y	
ReferenceDate	Datetime	Y	

#### Indexes

Name	Kind	Columns	Purpose
PK_SurveyReferences	P	SurveyReferences	Primary key.
IXSurveyReferencesTranNo		TranNo	To improve the running performance of the "Ticket Reason Report"

### 3.177 Surveys

The survey capability is a tool offering many features to ease and assist in the accumulation of customer survey data. The features of this function make it possible to fully self-define, administer and tally all or some results of various surveys. Because of the versatility of this function, users are able to pre-define literally thousands of unique surveys and multiple choice answer sets.

The **Surveys** table contains the information as to the frequency of a given survey should be asked to the customer and whether the response to the survey is required to complete the current transaction.

#### Columns

Column	Type	Allow Nulls	Description
SurveyUniqueID	Int	N	Primary key, always unique. System generated.
SurveyID	Int	N	User definable survey number. This is the value used to identify a survey throughout the system.
Name	Char(40)	Y	The name of the survey
Required	Bit	N	If TRUE, the user cannot escape out of the survey
Frequency	Int	N	The frequency is a feature when used in conjunction with the transaction prompt to control how often a particular survey is issued in ticketing mode.
ElapsedSendMinutes	Int	Y	Specifies the number of minutes after entering the attraction to wait before sending the survey
RequestWebTemplateID	Int	Y	Allows for specification of the web template to be used when sending the survey to the customer. FK to WebTemplates.WebTemplateID
PositiveResponseWebTemplateID	Int	Y	Allows for specification of the web template to be used when sending the follow up email when a positive survey is received. FK to WebTemplates.WebTemplateID
NegativeResponseWebTemplateID	Int	Y	Allows for specification of the web template to be used when sending the follow up email when a negative survey is received. FK to WebTemplates.WebTemplateID
PositiveSurveyThreshold	Int	Y	Percent that anything on or above will be considered a positive response to the survey.
SurveyGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSurveysSurveyID	P	SurveyID	Primary key.

### 3.178 Taxes

The **Taxes** contains the definition of the different taxes that can be applied to sales transactions in the system.

#### Columns

Column	Type	Allow Nulls	Description
TaxUniqueId	Int	N	Primary key, always unique.
Name	Char(40)	N	The Name is the label applied to the tax for on-screen viewing. Name does not affect the system operation.
TaxRate	Float	N	Rate is the percentage charged as tax. For example, 6% is entered as 6.0000.
FloorLimit	Float	Y	Non-Zero amount will indicate the floor limit for the individual tax.
TaxCap	Float	N	Cap is the maximum amount of the indicated tax charged per ticket. If a percentage value ends up less than the cap, the percentage is charged. If the percentage value exceeds the cap amount, the cap is charged.
Method	Char(1)	Y	The tax method to apply for the tax. <sup>1</sup>
TaxOnTax	Bit	Y	Indicates if taxable amount should include previous taxes
TaxID	Int	Y	BTree TaxID (from 1 to 8)
TaxGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTaxesTaxUniqueId	P	TaxUniqueId	Primary key.

#### <sup>1</sup> Method Values

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis

### 3.179 TaxTableHeaders

This table defines a tax table. Details for the tax table are in TaxTableDetails

#### Columns

Column	Type	Allow Nulls	Description
TaxTableHeaderUniqueID	Int	N	Primary key, always unique
TaxTableHeaderID	Int	N	Record ID
ExternalID	Char(10)	N	User ID for the table
Name	Char(40)	N	Name of the tax table
EffectiveDate	DateTime	N	Effective date for the table
TaxTableHeaderGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTaxTableHeadersID	P	TaxTableHeaderUniqueID	Primary Key.

### 3.180 TaxTableDetails

This table holds all the details for a particular tax table. Tax tables are defined in TaxTableHeaders.

#### Columns

Column	Type	Allow Nulls	Description
TaxTableDetailUniqueID	Int	N	Primary key, always unique
TaxTableDetailID	Int	N	Record ID
TaxTableHeaderID	Int	N	Header record for this detail
SaleAmount	money	N	Sale amount this detail is for
TaxID	Int	N	Tax id for this detail
TaxAmount	money	N	Amount of tax for this sale amount and tax ID
TaxTableDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTaxTableDetailsID	P	TaxTableDetailUniqueID	Primary Key.
IXTaxTableDetailsHeaderAmount	IX	TaxTableHeaderID, SaleAmount	Speed up look-ups

### 3.181 Templates

The Templates table stores Ticket type templates to assign discounted fares to function keys. This is the primary link to the rest of Galaxy because each ticket type includes the PLU item number for the Galaxy Items table.

#### Columns

Column	Type	Allow Nulls	Description
TemplateID	Int	N	Primary key, always unique
Carrier	Char(4)	Y	Refers to Carrier.Carriers
TemplateName	Char(24)	Y	Name of this ticket type template
Description	Char(50)	Y	Description

#### Indexes

Name	Kind	Columns	Purpose
PKTemplatesTemplateID	P	TemplateID	Primary Key
IXTemplatesCarrierTemplateName	I	Carrier, TemplateName	Index on Carrier and TemplateName
IXByTemplateName	I	TemplateName	Index on TemplateName

### 3.182 TemplateDetails

The TemplateDetails table stores up to 20 records representing F1 - F10 for one way and round trip tickets.

#### Columns

Column	Type	Allow Nulls	Description
TemplateDetailID	Int	N	Primary key, always unique
TemplateID	Int	N	Detail Key for Templates
FKey	SmallInt	Y	Function key 1 through 10
PWB	Char(1)	Y	P=POS only, W=Web only, B (or blank) = Both
Descr	Char(16)	Y	Brief description of this ticket type (mostly used on ticket coupons and reports)
LongDescr	VarChar(32)	Y	Description of this ticket type
FareClass	Char(2)	Y	User defined abbreviations for adult, child, etc
RouteType	Char(2)	Y	OW or RT for one way or round trip ticket type
Percentage	SmallInt	Y	Percent of base fare
Rounded	Char(1)	Y	Up, Down, Nearest
RoundTo	SmallInt	Y	5 is nickel, 10 is dime, 25 is quarter, etc
GalaxyItemPLU	Char(20)	Y	Refers to Items.PLU
ICWItemPLU	Char(20)	Y	The "In Conjunction With" item PLU

#### Indexes

Name	Kind	Columns	Purpose
PKTemplateDtlsTemplateDetailID	P	TemplateDetailID	Primary Key.
IXTemplateDetailsTemplateID	I	TemplateID	Index on TemplateID

### 3.183 TerminalIDs

TerminalIDs table contains the details for a credit authorization protocol. The details in this table are used when a credit authorization request is sent to the Galaxy's payment processor. Data for this table is defined under Galaxy>Data Maintenance>System Maintenance>Credit Authorization>Terminal IDs edit screen. Data defined in Galaxy can be exported to this SQL table using DBSync.

Column	Type	Allow Nulls	Description
TerminalID	Integer	N	Primary key. Obtained from GatewayCounters
Protocol	Varchar(30)	Y	Protocol for which TerminalIDs are being defined
TerminalID	Varchar(30)	Y	TerminalID for the given protocol
MerchantID	Varchar(30)	Y	MerchantID for the given protocol
PrimaryPhoneNo	Varchar(30)	Y	Primary phone number to use for given protocol
SecondaryPhoneNo	Varchar(30)	Y	Secondary phone number to use for given protocol
Name	Varchar(50)	Y	Name of the terminal
HostField1	Varchar(50)	Y	Host Field1 data
HostField2	Varchar(50)	Y	Host Field2 data
HostField3	Varchar(50)	Y	Host Field3 data

#### Indexes

Name	Kind	Columns	Purpose
PKTerminalIDsTerminalID	P	TerminalID	Primary Key.
IXTerminalIDsProtocol		Protocol	Index, used by query to retrieve details by the given protocol

### 3.184 TerminalConfig

The **TerminalConfig** table contains all SQL-based Terminal ID settings that are specific to individual stations. When using a credit card protocol that supports SQL-based Terminal ID settings, Galaxy automatically downloads this data.

#### Columns

Column	Type	Allow Nulls	Description
TerminalConfigID	Int	N	Primary key, always unique
Node	Int	N	Node number of the Galaxy station
Protocol	Char(10)	N	The name of the protocol <sup>1</sup>
Name	VarChar(20)	N	Name of configuration option <sup>2</sup>
Value	VarChar(128)	N	Value of configuration option

#### Indexes

Name	Kind	Columns	Purpose
PK_TerminalConfigID	P	TerminalsConfigID	Primary key.

#### <sup>1</sup> Protocol Values

Value	Gateway Constant Name	Description	Notes
HYPERCOM	HYPERCOM_SQL_PROTOCOL	Corresponds to the HYPERCOM1 and HYPERCOM2 protocols (Hypercom ISO 8583 authorization)	The Network International Identifier (NII) value for a station (corresponds to element 24) is stored in HostField1. HostField2 and HostField3 are not stored for this protocol. However, some Terminal ID fields (including Host ID #2) are assigned from values in the SettlementConfig table during Galaxy's Terminal ID download process.

#### <sup>2</sup> Name Values

Value	Gateway Constant Name	Description
TerminalID	TERMINALS_CONFIG_TERMINAL_ID	Assigned by credit card processor to identify the station performing credit transactions. Corresponds to "Terminal ID" field of Terminal ID record.
MerchantID	TERMINALS_CONFIG_MERCHANT_ID	Assigned by credit card processor to identify the merchant. Corresponds to "Merchant ID" field of Terminal ID record.
HostField1	TERMINALS_CONFIG_HOST_FIELD_1	Meaning varies depending on protocol. Corresponds to "Host ID" field of Terminal ID record.
HostField2	TERMINALS_CONFIG_HOST_FIELD_2	Meaning varies depending on protocol. Corresponds to "Host ID #2" field of Terminal ID record.
HostField3	TERMINALS_CONFIG_HOST_FIELD_3	Meaning varies depending on protocol. Corresponds to "Host ID #3" field of Terminal ID record.

### 3.185 TktRanges

The system allows a user to set up a range of ticket serial numbers that is considered valid for certain access codes.

When the Check Range option of an access code is set, the system checks the scanned serial number against a range of serial numbers that are considered valid. These ranges are defined by access code and stored in a separate database.

#### Columns

Column	Type	Allow Nulls	Description
TktRangeID	Int	N	Primary key, always unique. System generated.
AccessCode	Int	N	The access code value where the ticket is valid
RangeLo	char(20)	N	The lowest serial number value where the ticket is valid
RangeHi	char(20)	N	The highest serial number value where the ticket is valid
Comments	char(20)	N	The description of this ticket range

#### Indexes

Name	Kind	Columns	Purpose
PKTktRangesTktRangeID	P	TktRangeID	Primary key.

### 3.186 TktSets

A Ticket Set is a list of coupon format numbers that comprise the ticket. Each ticket type associated with a function key must be given a ticket set number for that ticket type to actually print a ticket. Up to 36 coupon numbers may be included in a ticket set. The system can store up to 999 ticket sets as determined under Set Max Ticket Sets.

The TktSets table contains the header ticket set definitions, and the TktSetDetails table contains the details of the ticket set.

#### Columns

Column	Type	Allow Nulls	Description
TktSetID	Int	N	Primary key, always unique. System generated.
TktSetNo	Int	N	User definable ticket set number. This is the value used to identify a ticket set throughout the system.
Descr	char(20)	N	The Description is a label that informs the user of the intended use of the ticket set.

#### Indexes

Name	Kind	Columns	Purpose
PKTktSetsTktSetID	P	TktSetID	Primary key.

### 3.187 TktSetDetails

Table to hold the Ticket Set Details. Detail table to the TktSets table, this table can have 1 to many entries associated to each entry in the TktSets table.

#### Columns

Column	Type	Allow Nulls	Description
TktSetDetailID	Int	N	Primary key, always unique.
TktSetNo	Int	N	Ticket number of the TktSet header record.
CoupldxNo	Int	N	The 'I-th' coupon in the TktSet.
CouponNo	Int	Y	Number of the 'I-th' coupon.
PrinterNo	Int	Y	Printer Number of the 'I-th' coupon.
PrimaryTray	Int	Y	The primary tray to feed tickets from for printers that support multiple trays.
SecondaryTray	Int	Y	The secondary tray to feed tickets from for printers that support multiple trays.

#### Indexes

Name	Kind	Columns	Purpose
PKTktSetDetailID	P	TktSetDetailID	Primary Key.
IXTktSetDetailsTktSetNoID	I	TktSetNo	Index on TktSetNo for use when loading details for a TktSet.

### 3.188 TicketCodes

The Galaxy application has ability to exchange a ticket for another item such as a pass or a different kind of ticket. However, if the ticket being exchanged was generated outside of Galaxy, a value needs to be linked in order to properly calculate the exchange values.

The **TicketCodes** database allows the user to set up ticket codes that correspond to ticket values and account numbers to be used for the exchange function. To properly configure this option, an Access Code must be set up to reference the *TicketCodes* table.

#### Columns

Column	Type	Allow Nulls	Description
TicketCodeID	Int	N	Primary key. Always unique. System generated.
TicketCode	Char(20)	N	A numeric code to be assigned to this ticket code. For proper operation of the system, this needs to match the access code defined for this particular ticket code.
Description	Char(20)	N	A description for this specific ticket code. This is displayed on the window when making an exchange.
Price	Money	N	The price for the ticket being represented by this ticket code. This ensures proper credit to the ticket holder for any exchanges.
Exchange	Bit	N	If FALSE, the ticket holder cannot exchange a ticket for another type.
Company	Int	N	The account number to be used for exchange values for this ticket code. The first three digits of this field are the company number, the next three are the category and the last two are the sub category.
Category	Int	N	As noted above.
SubCat	Int	N	As noted above.
EditPrice	Bit	N	If TRUE, allows the ticket seller to override the value of the ticket upon exchange.
TaxIncluded	Bit	N	If TRUE, the item in exchange includes taxes.
TaxFlags	Int	N	The taxes used for the ticket.

#### Indexes

Name	Kind	Columns	Purpose
PKTicketCodesTicketCodeID	P	TicketCodeID	Primary key.

### 3.189 TicketNotes

Stores user defined note information for a given ticket or pass. Multiple notes can be stored for each ticket and pass.

#### Columns

Column	Type	Allow Nulls	Description
TicketNoteID	Int	N	Primary key, always unique
VisualID	Varchar(40)	Y	Visual ID of the ticket/pass
TicketType <sup>1</sup>	Int	Y	0 is for Ticket 1 is for Pass
Description	Char(25)	N	Short Description of the note
Note	VarChar(255)	N	The note
UserID	Int	Y	User ID
Name	Char(25)	N	User name

#### Indexes

Name	Kind	Columns	Purpose
PKTicketNotesTicketNoteID	P	TicketNoteID	Primary Key.
IXTicketNotesVisualID	I	VisualID	Allow queries by visual id

<sup>1</sup> TicketType Values

Value	Gateway Constant Name	Description
0	ST_TICKET_TYPE	Ticket
1	ST_PASS_TYPE	Pass
2	ST_PACKAGE_TYPE	Package

### 3.190 TransactionalUpsellOptionRequirements

The TransactionalUpsellOptionRequirements table stores the requirements for displaying an upsell option based on a combination of items entered into a transaction. The item to offer for upsell is defined in the TransactionalUpsellOptions table.

#### Columns

Column	Type	Allow Nulls	Description
TransUpsellOptionReqUniqueID	Integer	N	Primary key, always unique.
TransactionalUpsellOptionRequirementID	Integer	N	Local key, unique
TransactionalUpsellOptionID	Integer	N	ID number of the transactional upsell option that the requirement is associated with. This is a foreign key to TransactionalUpsellOptions.TransactionalUpsellOptionID.
PLU	VarChar(20)	N	The PLU of the item that is required to be in the transaction for the upsell option to display.
MinQuantity	Integer	N	The minimum quantity of the requirement PLU that must be present in the transaction for the upsell option to be offered.
MaxQuantity	Integer	N	The maximum quantity of the requirement that will be replaced if the upsell option is selected.

#### Indexes

Name	Kind	Columns	Purpose
PKTransUpsellOptionReqUniqueID	P	TransUpsellOptionReqUniqueID	Primary key - Unique ID in the table.
IXTransUpsellOptionID	A	TransactionalUpsellOptionID	Used to load all requirements for an upsell option

### 3.191 TransactionalUpsellOptions

The TransactionalUpsellOptions table contains information about upsell options that are available based on a combination of items in a transaction.

#### Columns

Column	Type	Allow Nulls	Description
TransUpsellOptionUniqueID	Integer	N	Primary key, always unique.
TransactionalUpsellOptionID	Integer	N	Local key, unique
Name	VarChar(50)	N	The name used to identify the upsell option
UpsellPLU	VarChar(20)	N	This is the PLU that can be upsold to based on the items in the transaction.
SequenceNo	Int	Y	Allows for defining the order that the upsell options will be displayed in.
UpsellType	Int	Y	Defines the type of upsell this option is for. <sup>1</sup>
PictureID	Int	Y	FK to Pictures.PictureID, picture that can be displayed for this transactional upsell option.

#### Indexes

Name	Kind	Columns	Purpose
PKTransUpsellOptionUniqueID	P	TransUpsellOptionUniqueID	Primary key - Unique ID in the table.
IXTransUpsellOptionID	A	TransactionalUpsellOptionID	Local key - unique identifier in local database

<sup>1</sup> UpsellType Values

Value	Gateway Constant Name	Description
0	TRANSACTIONAL_UPSELLTYPE_REPLACE	Upsell is a replacement
1	TRANSACTIONAL_UPSELLTYPE_ADDON	Upsell is an add-on

### 3.192 TransportationTicketTypes

This table stores the bus ticket type including the price, route type and fare class.

#### Columns

Column	Type	Allow Nulls	Description
TransTicketTypeID	Int	N	Primary key, always unique
Fare	Currency	Y	Price of the ticket
TemplateID	Int	Y	Ticket type template name
Fkey	Int	Y	One to one with Function Key Number on POS
Descr	VarChar(24)	Y	Description, ADULT, CHILD, STUDENT
FareClass	Char(2)	Y	Type of fare, AD, CH, SR, ST
RouteType	Char(2)	Y	OW or RT
Percentage	Float	Y	Percent of base fare
Rounded	Char(1)	Y	U-up, D-down, or N-nearest
RoundTo	Float	Y	Round to in cents, .05 = nickel, .25 = quarter
GalaxyItemID	VarChar(20)	Y	Galaxy item id, entered by user
ICWPLU	VarChar(20)	Y	Plu for ICW item

#### Indexes

Name	Kind	Columns	Purpose
PKTransTktTypesTransTktTypeID	P	TransTicketTypeID	Primary Key.

### 3.193 UpsellOptions

This table contains the items that can be sold as replacement or in addition to the selected PLU. Initially this feature will only be supported for passes on the web store with replacement items.

#### Columns

Column	Type	Allow Nulls	Description
UpsellOptionID	Int	N	Primary key, always unique
PLUSource	Char(20)	N	PLU that is the starting point for the upsell transaction
PLUDestination	Char(20)	N	PLU associated with this option to be used if it is selected
Name	Varchar(50)	N	Name presented for upsell
Description	Varchar(217)	Y	Additional information about the upsell option
UpsellType <sup>1</sup>	Int	N	The Upsell type defines how to interpret the UpsellOption when selecting it for sale
SequenceNo	Int	Y	Index of the PLU destination
PictureID	Int	Y	FK to Pictures.PictureID, picture that can be displayed for this upsell option
ScriptTemplateID	Int	Y	Web template ID of the script to be used for this upsell option
Hide	Bit	N	Tells if this option should be hidden from users when upsell options are displayed, defaults to FALSE
Emphasis <sup>2</sup>	Int	Y	How this option should be emphasized when upsell options are displayed, defaults to 0
UpsellOptionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKUpsellOptionsUpsellOptionID	P	UpsellOptionID	Primary Key.
IXUpsellOptionsPLUSource	IX	PLUSource	The list of upsell options will always be retrieved by the PLUSource.
IXUpsellOptionsLastUpdate	IX	LastUpdate	Only created when Smart Upsell is enabled to help with dynamic configuration loading.

#### <sup>1</sup> UpsellType Values

Value	Gateway Constant Name	Description
0	REPLACEMENT_TYPE	The upsell defined in the PLUDestination will be used as a replacement to the PLUSource.
1	ADD_ON_TYPE	The upsell defined in the PLUDestination will be added to the transaction as an additional item to purchase.

#### <sup>2</sup> Emphasis Values

Value	Gateway Constant Name	Description
0	Normal	This option is listed as normal
1	Low	This option should be deemphasized
2	High	This option should be emphasized

### 3.194 UserProfileItemGroups

This table stores information that relates user profiles to item groups. Each row in the table will identify an item group that is valid for a user with the specified profile to sell.

#### Columns

Column	Type	Allow Nulls	Description
UserProfileItemGroupID	Int	N	Primary key, always unique. System generated.
UserProfileID	Int	Y	Foreign key to UserProfiles.UserProfileID
ItemGroupID	Int	Y	The item group ID number that this profile can sell
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKUserProfileItemGroups	P	UserProfileItemGroupID	Primary key.

#### <sup>1</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore UserProfileItemGroup records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store UserProfileItemGroups.

### 3.195 UserProfilePassKindGroups

This table stores information that relates user profiles to pass kind groups.

#### Columns

Column	Type	Allow Nulls	Description
UserProfilePassKindGroupID	Int (Identity)	N	Primary key, always unique
UserProfilePassKindGroupGUID	UniquedIdentifier	N	Alternate Primary Key
UserProfileID	Char(10)	Y	short name of the user profile from UserProfiles table (Name)
PassKindGroupGUID	UniquedIdentifier	N	Foreign key to PassKindGroups table (PassKindGroupGUID)

#### Indexes

Name	Kind	Columns	Purpose
PKUserProfilePassKindGroupID	P	UserProfilePassKindGroupID	Primary key

### 3.196 UserProfiles

This table stores information for the different user profiles that apply to the current users of the system. User profiles are used to enforce/prevent access to different modules, function, and areas of the system.

#### Columns

Column	Type	Allow Nulls	Description
UserProfileID	Int	N	Primary key, always unique. System generated.
Name	Char(10)	Y	User definable short name of the profile
Description	Char(20)	Y	User definable description for the profile
PasswordExpirationDays	Int	Y	The number of days a user of the specified profile has between password changes.
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>1</sup>
Publish	Bit	Y	If this is set, then the profile will be published
ExternalGroup	Nvarchar(255)	Y	Identifier used to match external group membership. Matched to a Common Name (CN) in LDAP
UserProfileGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKUserProfilesUserProfileID	P	UserProfileID	Primary key.
IXExternalGroup	IX	ExternalGroup	Speed up searching. Used during authentication

#### <sup>1</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore UserProfile records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store UserProfiles.

### 3.197 UserSettings

The UserSettings table is used to store configuration information for SQL only forms in the Galaxy application, such as Planner.

#### Columns

Column	Type	Allow Nulls	Description
UserSettingID	Int	N	Primary key, always unique
UserID	Int	Y	ID of the user who last changed this setting; Foreign key to GxUsers.UserID
LineNum	Int	Y	Sequence value that defines the order in which the records should be used/displayed
Section	VarChar(1024)	Y	The general grouping/category that the setting belongs to
KeyName	VarChar(1024)	Y	Name of the setting
Value	VarChar(1024)	Y	Current value of the setting

#### Indexes

Name	Kind	Columns	Purpose
PKUserSettingsUserSettingID	P	UserSettingID	Primary Key
IXUserSettingsUserSectionKey	IX	UserID, Section, KeyName	

### 3.198 ValueLinkConfig

This table is used by Galaxy for ValueLink configuration by node number.

#### Columns

Column	Type	Allow Nulls	Description
ValueLinkConfig	Int	N	Primary key, always unique
NodeNumber	Int	N	Unique Node identification number
MerchantID	Varchar(25)	N	Merchant identification number
AlternateMerchantID	Varchar(20)	N	
User1	NVarChar(20)	Y	An optional user-defined value that can be sent as part of the request to ValueLink.
User2	NVarChar(20)	Y	An optional user-defined value that can be sent as part of the request to ValueLink.
EchoBack	NVarChar(26)	Y	An optional user-defined value that can be sent as part of the request to ValueLink. This value is echoed back in the response from ValueLink.

#### Indexes

Name	Kind	Columns	Purpose
PKValueLinkConfigID	P	ValueLinkConfigID	Primary Key.

### 3.199 VenueLimits

Stores information about the System Defined Attributes.

#### Columns

Column	Type	Allow Nulls	Description
VenueLimitID	Int	N	Primary key, always unique
MaxGroupQty	Int	Y	Maximum number of groups allowed
MaxGuestQty	Int	Y	Maximum number of guests allowed

#### Indexes

Name	Kind	Columns	Purpose
PKVenueLimitsVenueLimitID	P	VenueLimitID	Primary Key.

### 3.200 VenueLimitTotals

Stores the total number of group and guests for each group visit date.

#### Columns

Column	Type	Allow Nulls	Description
VenueLimitTotalID	Int	N	Primary key, always unique
TotalGroupQty	Int	N	Maximum number of groups allowed
TotalGuestQty	Int	N	Maximum number of guests allowed
GroupVisitDate	DateTime	N	Group visit date for an order/customer.

#### Indexes

Name	Kind	Columns	Purpose
PKVenueLmtTtlVenueLmtTtlID	P	VenueLimitTotalID	Primary Key.

### 3.201 VisaNetTerminals

The **VisaNetTerminals** table is used to store terminal specific information for installations using the Visanet protocol for credit card authorizations. Non-Visanet installations will not use this table. All columns in the table (except NodeNo) are specified by the VisaNet interface specification and descriptions for each column can be found in the interface documentation.

#### Columns

Column	Type	Allow Nulls	Description
VisaNetTerminalID	Int	N	Primary key, always unique
NodeNo	Int	Y	The POS node number used when gathering data from this table. Each node that will authorize credit cards will have a row in this table. The settlement node will also have a row in this table and NodeNo will specify the node doing settlements.
TerminalID	Varchar(50)	Y	
AcquirerBin	Varchar(50)	Y	
AgentNumber	Varchar(50)	Y	
ChainNumber	Varchar(50)	Y	
MerchantNumber	Varchar(50)	Y	
StoreNumber	Varchar(50)	Y	
DeviceCode	Varchar(50)	Y	
IndustryCode	Varchar(50)	Y	
CurrencyCode	Varchar(50)	Y	
LanguageInd	Varchar(50)	Y	
TimeZone	Varchar(50)	Y	
CountryCode	Varchar(50)	Y	
CityCode	Varchar(50)	Y	
CategoryCode	Varchar(50)	Y	
MerchantName	Varchar(50)	Y	
MerchantLoc	Varchar(50)	Y	
MerchantState	Varchar(50)	Y	
MerchantLocNum	Varchar(50)	Y	
TermIDNum	Varchar(50)	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKVisaNetTerminalID	P	VisaNetTerminalID	Primary Key.

### 3.202 WaitLists

Wait Lists are records of contacts who want to purchase tickets to a sold out event. If tickets become available, they will be offered to the contacts on the Event wait list. (New Resource Management feature.)

#### Columns

Column	Type	Allow Nulls	Description
WaitListID	Integer	No	Primary key, always unique
EventID	Integer	Yes	Points to the Event the contact wants tickets for
ContactID	Integer	Yes	Points to the Contact who wants tickets for the Event; link to CustContacts.CustContactID
WaitNumber	Integer	Yes	The sequence number of the wait list request
PriorityCodeTableValueID	Integer	Yes	The selection of the user attribute, the value from the lookup table.
ActionCodeTableValueID	Integer	Yes	Indicates how the contact was contacted when tickets became available <sup>1</sup>
ResponseCodeTableValueID	Integer	Yes	Indicates the contacts response to the Action <sup>2</sup>
TicketQuantity	Integer	Yes	The number of tickets the customer is interested in purchasing.

#### Indexes

Name	Kind	Columns	Purpose
PKWaitListWaitListID	P	WaitListID	Primary key
IXWaitListContactID		ContactID	Used to search for Wait List by Contact ID
IXWaitListEventID		EventID	Used to search for Wait List by Event ID

<sup>1</sup> Action Values

Value	Gateway Constant Name	Description
0	ACTION_TAKEN_PHONE_ID	Contacted by phone
1	ACTION_TAKEN_FAX_ID	Contacted by Fax
2	ACTION_TAKEN_MOBILE_PHONE_ID	Contacted by mobile phone
3	ACTION_TAKEN_EMAIL_ID	Contacted by email
4	ACTION_TAKEN_MAIL_ID	Contacted by mail
5	ACTION_TAKEN_OTHER_ID	other

<sup>2</sup> Response Values

Value	Gateway Constant Name	Description
0	RESPONSE_PURCHASE_ID	Contact purchased tickets
1	RESPONSE_NO_ANSWER_ID	Contact did not respond
2	RESPONSE_NOT_INTERESTED_ID	Contact no longer interested
3	RESPONSE_OTHER_ID	other

### 3.203 Warnings

This table contains messages from Lead which can be triggered at a particular time of the day. These warnings can optionally be set to start Change Orders.

#### Columns

Column	Type	Allow Nulls	Description
WarningID	Int	N	Primary key, Always unique
WarningTime	Datetime	N	The time of the warning
Message	Char(40)	N	Warning message text
EnableChangeOrder	Bit	N	This flag is set to enable change orders when time
LeadChangeOrderCmd	Bit	N	This flag is set for only one warning in the table. It flags which warning message is to be displayed when a lead operator selects the allow change order function.
Valid	Bit	N	

#### Indexes

Name	Kind	Columns	Purpose
PKWarningsWarningID	P	WarningID	Primary Key, always unique.

### 3.204 WorkGroups

This table stores information regarding work groups which are groups of GxUsers.

#### Columns

Column	Type	Allow Nulls	Description
WorkGroupId	Int	No	Primary Key
Description	Varchar(256)	No	Name or description of the Work Group

#### Indexes

Name	Kind	Columns	Purpose
PKWorkGroupId	P	WorkGroupId	Primary Key.

### 3.205 WorkGroupDetails

This table stores information regarding the members of a Work Group.

#### Columns

Column	Type	Allow Nulls	Description
WorkGroupDetailID	Int	No	PrimaryKey
WorkGroupID	Int	No	Foreign Key to WorkGroups table.
OwnerId	Int	No	Link to GxUser.UserID
OwnerType	Int	No	Currently only one type: 0 = GxUser

#### Indexes

Name	Kind	Columns	Purpose
PK WorkGroupDetailID	P	WorkGroupDetailID	Primary Key.
IXWorkGroupDetailsOwner	IX	OwnerType, OwnerID	Used when querying for user Activity statistics for display on main menu.

### 3.206 ZipCodes

This table contains zip code information. It is used by **Addresses** for **Customers** and **CustCategories**.

#### Columns

Column	Type	Allow Nulls	Description
ZipCodeID	Integer	N	New primary key as in France, different cities can have the same zipcode
ZipCode	Char(16)	N	Primary key, always unique. Contains postal code or zip-code in the United States.
State	VarChar(40)	Y	The State or Province the postal code resides in.
City	Varchar(40)	Y	The City the postal code resides in.
County	Char(20)	Y	The County the postal code resides in.
DMACode	Char(10)	Y	Demographic Marketing of Zip Code Areas.
AreaCode	Char(10)	Y	Telephone Area Code
CountryCode	Char(2)	Y	New Field. Foreign key reference to Countries.CountryCode (that is an alternate key)

#### Indexes

Name	Kind	Columns	Purpose
PKZipCodesZipCodeID	P	ZipCodeID	Primary Key.
IXZipCodesZipCode		ZipCode	Index used for Zipcode lookups

## 4 Access Control

#### 4.1 AccessCodeGroupDetails

Stores the details associated with each access code group.

##### Columns

Column	Type	Allow Nulls	Description
AccessCodeGroupDetailID	Int	N	Primary key, Always Unique
GroupId	Int	N	Link to the Header Table
AccessCode	Int	N	Link to AccessCode

##### Indexes

Name	Kind	Columns	Purpose
PKAccessCodeGroupDetailID	P	AccessCodeGroupDetailID	Primary Key

## 4.2 AccessCodeGroups

Stores access code groups.

### Columns

Column	Type	Allow Nulls	Description
AccessCodeGroupID	Int	N	Primary key, Always Unique
GroupID	Int	N	Link to the Details Table
Name	Char(128)	N	Description of Access Code Group

### Indexes

Name	Kind	Columns	Purpose
PKAccessCodeGroupID	P	AccessCodeGroupID	Primary Key

### 4.3 AccessCodeOverrides

Access Code Overrides allow a new way of making changes to existing Access Codes without allowing users to access the Access Codes table (i.e. the privileges for editing Access Code Overrides are distinct from the privileges for editing Access Codes).

An Access Code Override consists of three fields: a *description* for end-user use, an *access code* and an *action*. An action is essentially a change to an access code setting. For example, the *Enable Biometric Activities* action enables biometric registration for an access code even if biometrics registration is currently disabled for that access code.

This table is populated by MWS when the *Edit Access Code Overrides* icon is selected in Access Control Maintenance. Access Code Overrides can be sent to all ACPS by invoking the *Send Access Code Overrides* icon in MWS.

#### Columns

Column	Type	Allow Nulls	Description
AccessCodeOverrideID	Int	N	Primary Key
Access Code	Int	N	Foreign key to AccessCodes.AccessCodeID
Descr	Char(40)	Y	A user-defined description for the row
Action	Int	N	An action to apply to the Access Code, see Action Values <sup>1</sup> below. <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKAccessCodeGroupID	P	AccessCodeGroupID	Primary Key
IXAccessCodeOverridesAccessCode	A	AccessCode	Search on Access Code

#### <sup>1</sup> Action Values

Value	Gateway Constant Name	Description
0	oaEnableBiometricActivities	Enable Biometric activities for the indicated Access Code. This is checked by the Ticket Validator in ACS32 when validating tickets. This setting overrides the value of AccessCodes.BiometricRegistrationMode.
1	oaDisableBiometricActivities	Disable Biometric activities for the indicated Access Code. This is checked by the Ticket Validator in ACS32 when validating tickets. This setting overrides the value of AccessCodes.BiometricRegistrationMode.

#### 4.4 AccessCodePrintDetails

The AccessCodePrintDetails table contains information about what to print when a ticket or pass is scanned at an ACP in the Galaxy Point of Sale. This table allows different things to print based on the facility at which the ticket is scanned..

##### Columns

Column	Type	Allow Nulls	Description
AccessCodePrintDetailUniqueID	Integer	N	Primary key, always unique.
AccessCodePrintDetailID	Integer	N	Local key, unique
AccessCode	Integer	N	The access code that will trigger printing. Foreign key to AccessCodes.AccessCode.
FacilityID	Integer	N	The facility at which the ticket set or PLU should be printed. A value of 0 indicates that this should be printed at all facilities.
Printer	Integer	N	The number of the printer that will print the ticket set after a valid scan. There can be any number of these configured for an access code and facility. This will not be used if the record is configured to print a PLU.
TktSetID	Integer	N	The ID of the ticket set that will print. There can be any number of these configured for an access code and facility. This will not be used if the record is configured to print a PLU.
PLU	VarChar(20)	N	PLU of the ticket that should print as a replacement or supplemental ticket after a valid scan. There can be only one PLU configured for an access code per facility. This will not be used if the record is configured to print a ticket set.
ReplaceTicket	Bit	N	Indicates if the printed ticket will replace the original ticket scanned. (Only used if printing a PLU and not a ticket set)
PrintUsageType	Integer	Y	Indicates how the usage will be recorded when the new tickets are printed (Only used if printing a PLU and not configured to replace the ticket) <sup>1</sup>

##### Indexes

Name	Kind	Columns	Purpose
PKACPrintDetailUniqueID	P	AccessCodePrintDetailUniqueID	Primary key - Unique ID in the table.
IXACPrintDetailID	A	AccessCodePrintDetailID	Local key - unique identifier in local database
IXACPrintDetailAccessCode	A	AccessCode	Key to enhance loading the list by access code

##### <sup>1</sup> PrintUsageType Values

Value	Gateway Constant Name	Description
0	TPU_ORIGINAL	Usage is recorded for the original ticket scanned. The new printed tickets do not have usage recorded.
1	TPU_SUPPLEMENTAL	Usage is recorded for the new tickets that are printed. The original ticket is added to the reentry table, but no usage is recorded.

## 4.5 AccessCodes

An Access Code is used to describe a type of ticket as it relates to the Access Control System in general. Other non-access-control properties of tickets are defined in the **Items** table. References to "the ticket" in the following section imply any record in the **Tickets** table (or the physical ticket it represents) whose Access Code has the same value as the Access Definition's. Some of the **AccessCodes** fields are used only on the local ACS2 or POS ACCESS.DAT file corresponding to this table; therefore, making changes to this SQL table only (without updating the local databases) may not have the desired effect.

### Columns

Column	Type	Allow Nulls	Description
AccessCodeID	Int	N	Primary key, always unique. System generated.
AccessCode	Int	N	User definable access code number. This is the value used to identify an access definition throughout the system.
Name	Char(30)	Y	Description of the definition
ReportGroup	Char(12)	Y	Name report group to include counts of this code in
ValidateTicket	Bit	N	Check for valid record in ticket database
MaxParty	Int	Y	The maximum number of people who may be admitted
MultiUse	Bit	N	Check and decrement ticket's RemainingValue
ManyUse	Bit	N	Ticket may be used any number of times, if valid
InitialValue	Int	Y	The maximum number of uses for mulit-use tickets (when a ticket is created, its RemainingValue field initially has this value)
Dynamic	Bit	N	Ticket added to Tickets table if not there already
CheckRange	Bit	N	Check TktRange table for valid serial number
CheckReentry	Int	Y	Check ReEntry table for previous use
SoundName	Char(12)	Y	Sound file to play if ticket is valid
LogonCard	Bit	N	Access Definition is used for <i>logon card</i> barcodes
ValidMsg	Char(80)	Y	Message to display after valid scan (overrides default)
SuppressLog	Bit	N	Do not record entry in Usage table for this code
Consignment <sup>1</sup>	Bit	N	Check and update Consignment table (the consignment table is supported in AX1180 but is not used in Galaxy)
Positive	Bit	N	Update Tickets table when sold, returned, or voided <sup>1</sup>
Pass	Bit	N	Check and update Pass table <sup>1</sup>
TicketCode	Bit	Y	Use ticket code database <sup>1</sup>
Lights	Char(8)	Y	Y/N flag array of turnstile lights to activate if valid
OncePerDay <sup>2</sup>	Char(2)	Y	Specifies whether tickets and passes are valid only once per day
FromAccount	Int	Y	Used by the system to compute deferred revenue - when a ticket with this access code is used, the revenue is "moved" from this COA entry (company*100000 + category*100 + sub_cat) into the ToAccount COA entry
ToAccount	Int	Y	Used by the system to compute deferred revenue - when a ticket with this access code is used, the revenue is "moved" from the FromAccount COA entry into this COA entry (company*100000 + category*100 + sub_cat)
ReportingCode	Int	Y	company*100000 + category*100 + sub_cat
PositiveDB	Char(12)	Y	Name of Positive file
PartyUnlock	Bit	Y	Unlock turnstile for each person in party
NextAccess	Int	Y	If the "Multiple Facility" option is not selected in the POS ACS.INI file, this access code can be used for creating an additional positive record with the new access code. If that access code definition has its own NextAccess value, yet another positive ticket record will be created, and so on. This was useful for creating "multiple facility" tickets prior to the addition of "Multi-Park" functionality.
SuperOverride	Bit	Y	Supervisor approval required to override, if configured
MediaID	Int	Y	Media definition ID (counter_id)
DateSpecific	Bit	Y	Date specific ticket sales
PromptMsg	Char(40)	Y	Message to display to check pass picture
UsagePunches	Char(20)	Y	Usage punch positions
EjectOnPrompt <sup>3</sup>	Char(2)	Y	Indicates whether or not ACS2 will eject tickets on a prompt
UseLargeFont	Bit	Y	Use large font for PromptMsg
CalendarID	Int	Y	Foreign key to CalendarHeaders.CalHeaderID, specifying a calendar describing validity rules for this access code, or 0 for none
BankHeaderID	Int	Y	Foreign key to BankHeaders.BankHeaderID
FreeEntryCalendarID	Int	Y	Foreign key to CalendarHeaders.CalHeaderID, specifying a calendar that can be used to allow a "free" entry
FreeEntryMaxParty	Int	Y	Number of guests that are allowed to enter with "free" entry
LockTurnstile	Bit	Y	Lock turnstile after each party member is admitted
AutoValidate	bit	Y	If 1, validation is performed automatically and immediately when a ticket or pass using this access code is sold
ValidationResult <sup>4</sup>	Int	Y	The usage status value that will always result from scanning a ticket with this access code
OpenGateOnSale	Bit	Y	Open vehicle barrier gate after selling this item
OpenGateOnUse	Bit	Y	Open vehicle barrier gate after scanning this item
BiometricRegistrationMode <sup>5</sup>	Int	Y	Mode to use when registering ticket with a Biometric or other Identification provider(pending support of that feature by the provider)
IdentifyAgainstTransaction	Bit	Y	True if identification should be applied against any ticket in the transaction (vs. the individual ticket)
PerformTestVerification	Bit	Y	True if a Test verification should be performed when this ticket is first registered with the biometric or identification provider (pending support of that feature by the provider)
EnforceVelocityOnAdmission	Bit	Y	This column will determine if the velocity checks will be activated on the event of an admission
EnforceVelocityOnReentry	Bit	Y	This column will determine if the velocity checks will be activated on the event of an reentry
EnforceVelocityOnCrossover	Bit	Y	This column will determine if the velocity checks will be activated on the event of an crossover
EnforceVelocityOnReadmission	bit	Y	True if velocity checking should be performed when a guest reenters a facility they have entered before, and the reentry is treated as an admission because of the rules of validation
AutoReplenish	Bit	Y	Indicates if auto replenish is turned on or off

ReplenishType <sup>6</sup>	Int	Y	Type of auto-replenish
ReplenishValue	Int	Y	The value that triggers replenish. This is either the number of remaining uses or the day of month.
UsageFloor	Int	Y	The floor limit that applies to any ticket with this access code
NoteID	Int	Y	Foreign Key, references Notes SQL table. The note is used to store the policy and procedures for replenish.
PrintPLU	Char(20)	Y	PLU of ticket that prints after a valid scan
ReplaceTicket	Bit	Y	New printed ticket will replace the original ticket scanned
TicketPrintingUsage	Int	Y	Indicates how usage will be recorded when tickets are printed upon a valid scan <sup>7</sup>
CannotSuspend	Bit	Y	If this option is not set (value = 0; default), it denotes the autoreplenished ticket can be set to a suspended mode where the ticket no longer automatically replenished.
PartyPromptOnlyWhenOffline	Bit	Y	If 1, Validator will display Party Prompt for Party tickets only when scan is performed in offline mode (e.g. Usage.UsageCondition = ucOFFLINE)
StoredValue	Bit	Y	Set if the access code references a stored value card for point redemption
AttributeValueGroupID	Int	Y	Links one to many entries in the AttributeValues table to this Access Code. This ID is generated from the GatewayCounters table and will be unique for a group of values
ShowPhotoPrompt	Bit	Y	Controls showing the validation prompt for unverified photos - default value is True
AdmitGuestOnRejectedPhoto	Bit	Y	Controls allowing guest admission if photo is rejected - default value is True
InvalidatePassOnRejectedPhoto	Bit	Y	Controls invalidating the pass when the photo is rejected
VelocityCheckPassbacks	Int	Y	The number of passbacks allowed during a velocity check period.
MembershipValidationMode	Int	Y	The mode to use when validating a joint member pass. <sup>8</sup>
InvalidMsg	NVarChar(80)	Y	Message to display after an invalid scan (overrides default).
NegativeSoundFile	NVarChar(12)	Y	Sound file to play if ticket is invalid.
PositiveWebDisplayFile	NVarChar(255)	Y	Path and filename of an HTML file to display on the ACS32 touchscreen when a valid scan occurs.
NegativeWebDisplayFile	NVarChar(255)	Y	Path and filename of an HTML file to display on the ACS32 touchscreen when an invalid scan occurs.
PositiveUseColors	Bit	Y	A flag to indicate that the colors specified on the access code should be used for the ACS32 display for valid scans.
NegativeUseColors	Bit	Y	A flag to indicate that the colors specified on the access code should be used for the ACS32 display for invalid scans.
PositiveTextColor	Int	Y	The color to use for the text on the ACS32 display for valid scans.
NegativeTextColor	Int	Y	The color to use for the text on the ACS32 display for invalid scans.
PositiveBackgroundColorFrom	Int	Y	The background color to be used on the ACS32 display for valid scans. Use in conjunction with PositiveBackgroundColorTo to create a gradient effect.
NegativeBackgroundColorFrom	Int	Y	The background color to be used on the ACS32 display for invalid scans. Use in conjunction with NegativeBackgroundColorTo to create a gradient effect.
PositiveBackgroundColorTo	Int	Y	The background color to be used on the ACS32 display for valid scans. Use in conjunction with PositiveBackgroundColorFrom to create a gradient effect.
NegativeBackgroundColorTo	Int	Y	The background color to be used on the ACS32 display for invalid scans. Use in conjunction with NegativeBackgroundColorFrom to create a gradient effect.
CreatePassBanks	Bit	Y	If set to 1, when a pass is sold pass banks for that pass will be inserted into the TicketBanks table.
ShowGeneralPurposePromptOnFacilityAdmission	Bit	Y	1 if general purpose prompt should be displayed during a facility admission. Default value is 1.
ShowGeneralPurposePromptOnFacilityReentry	Bit	Y	1 if general purpose prompt should be displayed during a facility reentry
ShowGeneralPurposePromptOnFacilityCrossover	Bit	Y	1 if general purpose prompt should be displayed during a facility crossover. Default value is 1.
ShowGeneralPurposePromptOnAttractionAdmission	Bit	Y	1 if general purpose prompt should be displayed during an attraction admission
ShowGeneralPurposePromptOnAttractionReentry	Bit	Y	1 if general purpose prompt should be displayed during an attraction reentry
ShowGeneralPurposePromptOnAttractionCrossover	Bit	Y	1 if general purpose prompt should be displayed during an attraction crossover
RequiresVehicle	Bit	Y	Indicates that a vehicle is required to be present for sale and validation if a vehicle detector loop system is installed.
SurveyID	Int	Y	Foreign key to SurveyFields.SurveyID. Represents the survey to display to a user while validating a ticket using this access code.
GuestMovementValidationMode	Int	Y	The logic to use when determining whether guest movement is valid or not. This is used for the guest movement validation feature in ACS32. <sup>9</sup>
AttractionOnly	Bit	Y	If False (0), this entity (ticket, pass, debit card) can be positively validated at all ACPs (default setting). If True (1), this entity can be positively validated only at ACPs that are entrances to an Attraction. This is used to support scenarios where a ticket holder is given an additional ticket to scan at specific attractions within a facility. Those additional tickets would use the AttractionOnly setting. When a ticket is configured with AttractionOnly = 1, the facility's in-park count is not increased when that ticket is scanned at an attraction within that facility.
Inactive	Bit	N	True if access code is Inactive, and not visible on most picklists.
GPPFacilityAdmissionOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first facility admission
GPPFacilityReentryOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first facility reentry of the day
GPPFacilityCrossoverOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first facility crossover of the day
GPPAttractionAdmissionOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first attraction admission of the day
GPPAttractionReentryOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first attraction reentry of the day
GPPAttractionCrossoverOncePerDay	Bit	Y	1 if the general purpose prompt should only be displayed during the first attraction crossover of the day
ReprintOnFirstScan	Bit	Y	1 if the product should be reprinted (same PLU) when scanned first time at the ACP with the printer

**Indexes**

Name	Kind	Columns	Purpose
PKAccessCodesAccessCodeID	P	AccessCodeID	Primary key.
IXAccessCodesName		Name	Unknown.
IXAccessCodesAccessCode		AccessCode	Unique index to enforce that access code numbers are not duplicated.

<sup>1</sup> No more than one type of table update (positive, pass, consignment, or ticket code) should be selected.<sup>2</sup> OncePerDay Values

Value	Description
-------	-------------

Y	Specifies that a ticket can be used only once per day
N	Specifies that a ticket can be used multiple times per day, without the ACS2 system tracking reentry usages
R	Specifies that a ticket can be used multiple times per day, with the ACS2 system tracking reentry usages (with a warning)

**3 EjectOnPrompt Values**

Value	Description
N	No ticket ejection
I	Eject ticket from Inlet
O	Eject ticket from Outlet

**4 ValidationResult Values**

Value	Gateway Constant Name	Description
0	TICKET_VALID	The ticket was valid and accepted for admission.
6	TICKET_REENTRY	The ticket was found in the ReEntry table.
23	TICKET_CROSSOVER	The entry was a valid crossover.

**5 Status Values**

Value	Gateway Constant Name	Description
0	brmNONE	Do not perform Biometric registration
1	brmREGISTER_UPON_FIRST_SCAN	Register ticket with provider the first time it is scanned at ACS32 or Admission Control at POS (if applicable)

**6 ReplenishType Values**

Value	Gateway Constant Name	Description
0	AUTO_REPLENISH_BYUSES	Replenish on remaining uses
1	AUTO_REPLENISH_BY_EXPIRATION	Replenish by day of the month

**7 TicketPrintingUsage Values**

Value	Gateway Constant Name	Description
0	TPU_ORIGINAL	Usage is recorded for the original ticket scanned. The new printed tickets do not have usage recorded.
1	TPU_SUPPLEMENTAL	Usage is recorded for the new tickets that are printed. The original ticket is added to the reentry table, but no usage is recorded.

**8 MembershipValidationMode Values**

Value	Gateway Constant Name	Description
0	mvmUseACSConfig	Use the value defined in the ACS configuration.
1	mvmSingleCard	A single card represents all joint members. When that card is scanned at an ACP, the Info Pane will display details for all relationships associated with the primary contact for the pass and ACS will display joint member selection prompts. This value overrides any setting configured for the ACS configuration.
2	mvmCardPerMember	A card has been issued per joint member. When that card is scanned at an ACP, the Info Pane will only display details for the joint member associated with the card and ACS will not display joint member selection prompts. This value overrides any setting configured for the ACS configuration.

**9 GuestMovementValidationMode Values**

Value	Gateway Constant Name	Description
0	gmvUseACSConfig	Defer to ACPS.ValidateGuestMovements setting (default setting)
1	gmvValidate	Always validate guest movements at this ACP
2	gmvDoNotValidate	Never validate guest movements at this ACP

## 4.6 ACPS

An ACP, or Access Control Point, is a location at which a ticket or other barcoded media may be used for admission or redemption of value. An Access Control Point differs from a node in two ways. First, an ACP is a location which must be involved in Access Control. A node is any computer running Gateway software. Second, a node may contain more than one ACP. In many implementations, several ACP terminals are run from a single node.

### Columns

Column	Type	Allow Nulls	Description
ACPUniqueId	Int	N	Primary key, always unique. System generated.
ACPID	Int	N	User definable ACP ID number. This is the value used to identify an ACP throughout the system.
Name	Char(30)	N	Description of the Access Control Point.
Node	Int	N	Node of the computer to which this ACP belongs.
FacilityID	Int	N	Foreign key to Facility.IDNo, identifying the facility to which this ACP belongs.
Kind	Integer	Y	Kind of ACP, see table. <sup>1</sup>
ACPGroupID	Int	Y	Foreign key link to ACPGroups.ACGroupID. Specifies group that the ACP is a member of.
Points	Int	Y	The points that the ACP should remove from a Stored Value entity when it is scanned
CheckpointID	Int	Y	The corresponding check point ID for this ACP. Only applies when using Galaxy's interface to Handshake with auto-validation.
LayoutID	Int	Y	The corresponding layout ID for this ACP. Only applies when using Galaxy's interface to Handshake with auto-validation.
ExitFacilityID	Int	Y	This ACP exits the specified FacilityID. Foreign key to Facility.IDNo.
AttractionID	Int	Y	This ACP enters the specified AttractionID. Foreign key to Attractions.IDNo.
ValidateGuestMovements	Bit	Y	If False (0), guest movements should not be validated at this ACP (default setting). If True (1), guest movements should be validated at this ACP. Used for the guest movement validation feature in ACS32. This setting can be overridden by specific AccessCodes.GuestMovementValidationMode settings.

### Indexes

Name	Kind	Columns	Purpose
PKACPsACPUniqueId	P	ACPUniqueId	Primary key.
IXACPsFacilityID		FacilityID	To avoid time out when selecting the facility under Access Control Reports.
IXACPsACPID		ACPID	This unique index is specifically used by TCON32 when joining ACPS and Usage to load counts at startup. TCON32 startup time is drastically slower without this index.

### <sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
0	ACP_KIND_ACSP	Traditional ACS2 or POS Access Control Point
1	ACP_KIND_HANDHELD	Handheld Access Control Point

## 4.7 ACPGroups

The ACPGroups table contains definitions for ACP groupings. Multiple ACPs can belong to a group and a group can belong to one facility.

### Columns

Column	Type	Allow Nulls	Description
ACPGroupID	Int	N	Primary key, always unique
Description	VarChar(40)	Y	The group description
FacilityID	Int	Y	A link to the Facility table (Facility.IDNo). Specifies which facility the group is in.
ExitFacilityID	Int	Y	ACPs in this ACP Group exit the specified FacilityID. Foreign key to Facility.IDNo.
AttractionID	Int	Y	ACPs in this ACP Group enter the specified AttractionID. Foreign key to Attractions.IDNo.
ValidateGuestMovements	Bit	Y	If False (0), guest movements should not be validated at this ACP (default setting). If True (1), guest movements should be validated at this ACP. Used for the guest movement validation feature in ACS32. This setting can be overridden by specific AccessCodes.GuestMovementValidationMode settings.

### Indexes

Name	Kind	Columns	Purpose
PKACPGroupID	P	ACPGroupID	Primary Key.

## 4.8 ACP Modes

This table stores information about each ACP Mode defined in the system.

### Columns

Column	Type	Allow Nulls	Description
ACPModeID	Int	N	Primary key, always unique.
Name	Char (20)	Y	Name of ACP Mode.
Abbr	Char (8)	Y	Abbreviation of ACP Mode.
LoggedOnGuestDisplay	Varchar (200)	Y	Text to display on customer display when logged on.
LoggedOffGuestDisplay	Varchar (200)	Y	Text to display on customer display when logged off, or users are not allowed.

### Indexes

Name	Kind	Columns	Purpose
PKACPModesACPModeID	P	ACPModeDetailID	Primary Key.

## 4.9 ACPModeDetails

This table stores the settings, one row per setting, for each activated setting in an ACP Mode.

### Columns

Column	Type	Allow Nulls	Description
ACPMODEDetailID	Int	N	Primary key, always unique.
ACPMODEID	Int	N	Foreign key reference to ACPModes.ACPMODEID
Setting	Int	Y	Setting <sup>1</sup>
State	Int	Y	State this setting is used in <sup>2</sup>

### Indexes

Name	Kind	Columns	Purpose
PKACPMODEDetailsACPMODEDetailID	P	ACPMODEDetailID	Primary Key.

### <sup>1</sup> Setting Values

Value	Gateway Constant Name	Description
0	ACP_TS_CLOSED	Counting mode: not counting.
1	ACP_TS_ENTRY	Counting mode: counting entry rotations.
2	ACP_TS_EXIT	Counting mode: counting exit rotations.
3	ACP_TS_REENTRY	Counting mode: counting reentry rotations.
4	ACP_TS_ENTRY_EXIT	Counting mode: counting entry/exit rotations.
5	ACP_TS_REENTRY_EXIT	Counting mode: counting re-entry/exit rotations.
50	SCANS_ACCEPTED	Accept scans at this ACP.
51	ADMISSIONS_ACCEPTED	Accept Admission tickets at this ACP.
52	REENTRIES_ACCEPTED	Accept Reentry tickets at this ACP.
53	CROSSOVERS_ACCEPTED	Accept Crossover tickets at this ACP.
100	LOCK_ENTRY_TURNBAR	Lock turn-bar in entry direction at this ACP.
101	LOCK_EXIT_TURNBAR	Lock turn-bar in exit direction at this ACP.
102	TURNBAR_DOWN	The turnbar is down in this mode.
150	USERS_ALLOWED	Allow users to logon at this ACP.

### <sup>2</sup> State Values

Value	Gateway Constant Name	Description
0	NO_STATE	No state - not applicable (this should not be used in the table).
1	LOGOFF_STATE	This row is used to store a setting used when a user is not logged on.
2	LOGON_STATE	This row is used to store a setting used when a user is logged on.
3	NO_USERS_STATE	This row is used to store a setting for a mode that does not allow users.

## 4.10 AcpStatus

Each Access Control Point may be in one of several states. The **ACPStatus** table maintains the last known status of each ACP. In addition to the regular status, each record contains a turnstile counting mode column used for ACPS wired to turnstile switches. The turnstile mode determines what AccessCode to record in **Usage** table entries when counts are recorded.

### Columns

Column	Type	Allow Nulls	Description
AcpID	Int	N	Primary key, always unique.
Status	Int	N	A number relating to the ACP's current status. <sup>1</sup>
TsMode	Int	N	A number relating to the ACP's turnstile count mode. <sup>2</sup>
BiometricsDisabled	Bit	Y	True if Biometrics has been disabled on the ACP by specific user action (i.e., by invoking the Disable Biometrics function).
BiometricsRate	Int	Y	The rate at which biometric identifications occur. 0 = every scan, 1 = every scan, 5 = every fifth scan.

### Indexes

Name	Kind	Columns	Purpose
PKACPStatusACPID	P	ACPID	Primary key.

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	STATION_NEW	Uninitialized (the status has not yet been assigned).
1	STATION_OFFLINE	Offline (the station is not connected to the network).
2	STATION_LOGGED_ON	Logged On.
3	STATION_LOGGED_OFF	Logged Off.
4	STATION_CLOSED	Closed.

### <sup>2</sup> TsMode Values

Value	Gateway Constant Name	Description
0	ACP_TS_CLOSED	Not counting.
1	ACP_TS_ENTRY	Counting entries.
2	ACP_TS_EXIT	Counting exits.
3	ACP_TS_REENTRY	Counting re-entries.
4	ACP_TS_ENTRY_EXIT	Counting entries and exits.
5	ACP_TS_REENTRY_EXIT	Counting re-entries and exits.

## 4.11 ACSReports

The **ACSReports** table stores Access Definition groupings used in Access Control reports. Each record in the table defines either a report, a group of access codes in a report, or a range of access codes in a group.

### Columns

Column	Type	Allow Nulls	Comments
AcsReportID	Integer	N	Primary key, always unique.
Kind	Integer	N	A number representing the type of record as defined in Report Record. <sup>1</sup>
Name	Char(40)	N	A description of the report or report group which this record defines.
ReportAbbr	Char(12)	N	An abbreviation of either this report or this group's report, depending on Kind.
GroupAbbr	Char(12)	N	An abbreviation of this group or this range's group, depending on Kind.
FromCode	Integer	N	The first AccessCode to include in this group.
ThruCode	Integer	N	The last AccessCode to include in this group.
Seq	Integer	Y	Sequence number to sort group by (used for ordering groups on the report).
Style	Integer	Y	Specifies style of report. <sup>2</sup>
Subtract	Bit	Y	If selected, system will subtract quantity from rolling total.
TotLabel	Char(40)	Y	Label of group total line, if replacing the standard totals label.

### Indexes

Name	Kind	Columns	Purpose
PKACSReportsAcsReportID	P	AcsReportID	Primary key.

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	REPORT_REC_KIND	The record defines a report.
2	GROUP_REC_KIND	The record defines a group within a report.
3	ACCESS_REC_KIND	The record defines a range of access codes in a report group.

<sup>2</sup> Style Values

Value	Gateway Constant Name	Description
0	ORIGINAL_STYLE	Report uses original format.
1	RUNNING_STYLE	Report uses "running" format.

## 4.12 ACSTotals

The **ACSTotals** table contains the activity of the AccessCodes for a node. This is a manual process where the user select to load the usage data to view the access code activity for a given time frame.

### Columns

Column	Type	Allow Nulls	Description
ACSTotalID	Int	N	Primary key, always unique. System generated.
AccessCode	Int	N	The AccessCode of the activity.
AccessCodeDate	Int	N	The date of the AccessCode activity.
FromTime	Int	N	Starting time of the activity.
ThruTime	Int	N	Ending time of the activity.
Used	Int	Y	Number of time the AccessCode has been used.
Admits	int	Y	Number of the AccessCode which have been admitted to the park.

### Indexes

Name	Kind	Columns	Purpose
PKACSTotalsACSTotalID	P	ACSTotalID	Primary key.
IXACSTotalsAccessCodeDateTime		AccessCode, AccessDate, FromTime, ThruTime	

### 4.13 AdmissionHeaders

The **AdmissionHeaders** table is used to group together records in the **AdmissionDetails** table to form groups of operations (for Multi-park functionality).

#### Columns

Column	Type	Allow Nulls	Description
AdmissionHeaderID	Int	N	Primary key, always unique. System generated.
AdmHeaderID	Int	N	User definable admission list number. This is the value used to identify an admission list throughout the system.
Description	Char(50)	Y	Contains description of admission group.

#### Indexes

Name	Kind	Columns	Purpose
PKAdmHeadersAdmHeaderID	P	AdmissionHeaderID	Primary key.

#### 4.14 AdmissionDetails

The **AdmissionDetails** table stores the Operations selected for an Admission. Each record in this table belongs to a record in the **AdmissionHeaders** table.

##### Columns

Column	Type	Allow Nulls	Description
AdmDetailID	Int	N	Primary key, always unique.
AdmHeaderID	Int	N	Foreign key to AdmissionHeaders.AdmHeaderID.
OperationID	Int	Y	Foreign key to Operations.OperationID.

##### Indexes

Name	Kind	Columns	Purpose
PKAdmDetailsAdmDetailID	P	AdmDetailID	Primary key.

## 4.15 Attendance

This table stores attendance data (number of entries, reentries, exits and crossovers) for each datetime interval and for each facility.

### Columns

Column	Type	Allow Nulls	Description
Interval	datetime	N	This field plus FacilityID composes the Primary key that is always unique. This field contains the END of the interval (exclusive).
FacilityID	Int	N	Foreign key to Facility.IdNo. Interval plus this field composes the Primary key that is always unique.
Entries	Int	Y	Number of entries in this interval for this facility.
ReEntries	Int	Y	Number of reentries in this interval for this facility.
Exits	Int	Y	Number of exits in this interval for this facility.
Crossovers	Int	Y	Number of crossovers in this interval for this facility.
UpToDate	Bit	N	<p>There are some situations where the data in this record would be out-of-date (UpToDate = False):</p> <ul style="list-style-type: none"> <li>- As TCON updates the record each n seconds, if after the last update a new usage is registered and TCON crashes before a new update, the last record is out-of-date. When TCON goes back it sets the UpToDate field for this record to False.</li> <li>- Also if TCON stays offline for a while, and when it comes back it receives late updates from a previous day, it does not have the previous day data in memory to update the record. Then it just sets the interval as out-of-date (UpToDate = False).</li> </ul> <p>To refresh outdated records (UpToDate = False) the user should run the stand-alone ResetAttendance application. After running it, all records in the specified interval will be up-to-date (UpToDate = True).</p>

### Indexes

Name	Kind	Columns	Purpose
PKAttendanceIntervalFacilityID	P	Interval, Facility	Primary key.

## 4.16 AttractionCounts

The AttractionCounts table contains a record of the total Entries, Reentries and Exits for each Attraction.

### Columns

Column	Type	Allow Nulls	Description
AttractionID	Int	N	Primary key, Always Unique
Date	DateTime	N	Working date for the counts
EntryCounts	Int	N	Number of entries that have occurred
ReentryCounts	Int	N	Number of reentries that have occurred
EntryClicks	Int	N	Number of entry clicks recorded if a Turnstile click mechanism is installed
ReentryClicks	Int	N	Number of reentry clicks recorded if a Turnstile click mechanism is installed
ExitClicks	Int	N	Number of exit clicks recorded if a Turnstile click mechanism is installed

### Indexes

Name	Kind	Columns	Purpose
PKAttractionCountsAttrIDDate	P	AttractionID, Date	Primary Key

## 4.17 Attractions

Each attraction belongs to a Facility. An ACP can be configured to enter an Attraction.

### Columns

Column	Type	Allow Nulls	Description
AttractionID	Int	N	Primary key, Always Unique
IDNo	Int	N	Unique ID for end-user use, also unique ID in local database
Name	nvarchar(20)	Y	Shorter name of Attraction, displayed in various places
Description	nvarchar(100)	Y	Longer description of Attraction
FacilityID	Int	Y	The facility that this attraction belongs to. Foreign key to Facility.IDNo

### Indexes

Name	Kind	Columns	Purpose
PKAttractionID	P	AttractionID	Primary Key
PKAttractionIDNo		IDNo	Unique index

## 4.18 AttributeDefinitions

This table holds all the information pertaining to an attribute.

### Columns

Column	Type	Allow Nulls	Description
AttributeDefinitionID	Int	N	Primary key, always unique
Name	VarChar(30)	N	Name of Attribute
Description	VarChar(256)	N	Description of the Attribute
EntryType	Int	N	The Type of Entry used for this Attribute
CodeTableID	Int	N	ID of CodeTable to be used as default
Active	Boolean - Bit	N	Is the Attribute Active. 0 = Active, 1 = Inactive
AttributeCategoryID	Int	Y	FK to AttributeCategories table
ValueLength	Int	Y	The maximum length of the value of a free form attribute
DecimalPlaces	Int	Y	The maximum of decimal places a numeric attribute can have
AttributeDefinitionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKAttrDefsAttrDefID	P	AttributeDefinitionID	Primary Key
IXAttributeDefinitionsName	IX	Name	Disallow duplicate names

### <sup>1</sup> EntryType Values

Value	Gateway Constant Name	Description
0	aetNone	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
1	aetCodeTable	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
2	aetFreeForm	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
3	aetDate	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
4	aetNumeric	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
5	aetCurrency	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)
6	aetSubCodeTable	TattributeEntryType = (aetNone, aetCodeTable, aetFreeForm, aetDate, aetNumeric, aetCurrency, aetSubCodeTable)

## 4.19 AttributeCategories

Used to create categories for organizing attribute definitions.

### Columns

Column	Type	Allow Nulls	Description
AttributeCategoryID	Int	N	Primary key, always unique
Name	VarChar(30)	N	Name of attribute category
Active	Bit	N	Indicates whether this category is currently active
AttributeCategoryGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKACAttributeCategoryID	P	AttributeCategoryGroupID	Primary Key.

## 4.20 AttributeDefinitionGroups

This table connects attribute definitions to different connection kinds (Events, Event Types, Resources, Resource Types, Customer Categories, and Customers)

### Columns

Column	Type	Allow Nulls	Description
AttributeDefinitionGroupID	Int	N	Primary key, always unique
AttributeGroupID	Int	N	Identifier for a group of attributes (Unique for a grouping, ID comes from Gateway Counters)
AttributeDefinitionID	Int	N	Unique ID of the Attribute Definition
Required	Boolean - Bit	N	Determines if it is required
AttributeDefinitionGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKAttrDefGrpsAttrDefGrpID	P	AttributeDefinitionGroupID	Primary Key.
IXAttrDefGrpsAttrGrpID	IX	AttributeGroupID	Speed up searches, there can be many of the same ID

## 4.21 AttributeValues

This table holds values for attributes as well as connections between tables.

### Columns

Column	Type	Allow Nulls	Description
AttributeValueID	Int	N	Primary key, always unique
AttributeDefinitionID	Int	N	Links AttributeValue to a Attribute Definition
AttributeValueGroupID	Int	N	Links AttributeValue to Group of Values. This ID is generated from GatewayCounter and will be unique for a group of Values
Value	VarChar(256)	N	Attribute Value if freeform entry if CodeTableValueID = 0
CodeTableValueID	Int	Y	Links AttributeValue to a Code Table Value
Sequence	Int	Y	Sequence is a zero based integer value indicating the order of the attribute value in a list of attributes.
AttributeValueGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKAttrValuesAttributeValueID	P	AttributeValueID	Primary Key.
IXAttrValuesAttrDefID	A	AttributeDefinitionID	Speed up queries involving the attribute definition.
IXAttrValuesAttrValGrpID	A	AttributeValueGroupID	Speed up queries involving the attribute value group.

## 4.22 BankHeaders

The **BankHeaders** table contains bank header definitions, which are used to group together records in the **BankDetails** table. These two tables are used by TCON32 to populate the **TicketBanks** table with multi-park validation information for each ticket.

### Columns

Column	Type	Allow Nulls	Description
BankHeaderUniqueID	Int	N	Primary key, always unique. System generated
BankHeaderID	Int	N	User definable bank number. This is the value used to identify a bank throughout the system.
Description	Char(50)	Y	Bank Header description.
LimitCrossovers	Bit	Y	True if Crossovers should be limited when validating banks with this BankHeaderID

### Indexes

Name	Kind	Columns	Purpose
PKBankHeadersBankHeaderUniqID	P	BankHeaderUniqueID	Primary key.

## 4.23 BankDetails

The **BankDetails** table contains bank definition detail records belonging to a record in the **BankHeaders** table. These two tables are used by TCON32 to populate the **TicketBanks** table with multi-park validation information for each ticket.

### Columns

Column	Type	Allow Nulls	Description
BankDetailID	Int	No	Primary key, always unique.
BankHeaderID	Int	No	Foreign key to BankHeaders.BankHeaderID.
BankNo	Int	Yes	The detail number within the header (the first detail is bank number one, and so on).
InitialValue	Int	Yes	The initial value associated with this bank detail (this value can indicate "uses" but not necessarily).
Dependency	Int	Yes	The BankNo of the BankDetail record this detail is dependent on.
BankType <sup>4</sup>	Int	Yes	Defines how the bank is used.
Frequency <sup>1</sup>	Int	Yes	Indicates whether bank detail is once-per-day, reentry or crossover.
AdmissionID	Int	Yes	Foreign key to AdmissionHeaders.AdmHeaderID.
CalendarID	Int	Yes	Foreign key to CalendarHeaders.CalHeaderID.
FirstUseBasis <sup>2</sup>	Int	Yes	Defines how Relative to First Use calendars defined on the Bank are validated.
Points	Integer	Yes	The number of points a stored value account is given to start
Description	Varchar(128)	Yes	An optional description for the Bank Detail
RequirementBank	Int	Yes	A non-zero bank that RequirementKind is applied to when validating the bank
RequirementKind	Int	Yes	The kind of requirement applied to RequirementBank; values below <sup>3</sup>
RequirementMessage	Varchar(128)	Yes	The message to display if the RequirementBank condition is not satisfied when validating the bank
ExcludeFromHistory	Bit	Yes	Enabled if the "Ignore usages of this bank when admitting guests with another bank" option in the bank detail configuration is enabled.
ValidationPriority	Int	Yes	If non-zero, specifies the order that the bank is validated by the Admission Control Validator. Banks are validated in order of BankNo if ValidationPriority is not specified for each bank. ValidationPriority can be changed at any time unlike BankNo which is stored in the TicketBanks table.
Inactive	Bit	Yes	If False (0), this bank detail is active and should be examined during validation (default setting). If True (1), this bank detail is not active and should not be examined during validation. This is used to support scenarios where a specific bank detail is no longer in use.
ValidateUsesAndDaysMethod <sup>5</sup>	Int	Yes	How validation is handled in ACS for comparing remaining uses to remaining days on a ticket.
ValidateUsesAndDaysPrompt	nvarchar(128)	Yes	Message displayed when uses exceed days and using the 'Prompt the User' method for ValidateusesAndDaysMethod.
IsOverrideCaptureBank	Bit	No	If True (1), any ticket that references a bank set that includes this bank detail, when overridden, will record the override against this bank's balance. Only one "override capture" bank detail is allowed per bank set, and override capture banks are not validated in the traditional manner; they are only used for overrides.
AdditionalDependency	Int	Yes	The BankNo of the BankDetail record this detail is also dependent on.

### Indexes

Name	Kind	Columns	Purpose
PKBankDetailsBankDetailID	P	BankDetailID	Primary key.

### <sup>1</sup> Frequency Values

Value	Constant Name	Description
1	FONCE_PER_DAY	Once Per Day.
2	FREEENTRY	Reentry.
3	FCROSSOVER	Crossover.
4	FADMISSION_ONLY	Admission Only

### <sup>2</sup> FirstUseBasis Values

Value	Constant Name	Description
0	FIRST_USE_EXPIRATION	Use legacy mode
1	FIRST_USE_FROM_TICKET	Use the first use date of the ticket when validating the Calendars
2	FIRST_USE_FROM_BANK	Use the first use date of the bank when validating the Calendars

### <sup>3</sup> RequirementKind Values

Value	Constant Name	Description
0	brkAtLeastOneUse	Require RequirementBank to be used at least once
1	brkAllUses	Require RequirementBank to be completely used

### <sup>4</sup> BankType Values

Value	Constant Name	Description
1	ADMISSIONS_BANK_TYPE	Bank is used for Admission
2	POINTS_BANK_TYPE	Bank is used for Points

### <sup>5</sup> ValidateUsesAndDaysMethod Values

Value	Constant Name	Description
0	udDoNotCheck	Do not validate remaining uses against remaining days for this bank
1	udDoNotAllow	Validate remaining uses against remaining days for this bank, when uses exceed days, treat as invalid

2	udPromptUser	Validate remaining uses against remaining days for this bank, when uses exceed days, prompt as to whether or not to fail validation
---	--------------	---

## 4.24 CalendarHeaders

The **CalendarHeaders** table is used to group together validity rules (stored in the **CalendarDetails** table) to form "calendars".

### Columns

Column	Type	Allow Nulls	Description
CalendarHeaderID	Int	N	Primary key, always unique. System generated.
CalHeaderID	Int	N	User definable calendar number. This is the value used to identify a calendar throughout the system.
Description	Char(50)	Y	Description of calendar
Inactive	Bit	N	True if calendar is Inactive, and not visible on most picklists.

### Indexes

Name	Kind	Columns	Purpose
PKCalendarHeadersCalHeaderID	P	CalendarHeaderID	Primary key.

## 4.25 CalendarDetails

The **CalendarDetails** table contains rules to specify when a calendar record (in the **CalendarHeaders** table) is valid.

### Columns

Column	Type	Allow Nulls	Description
CalDetailID	Int	N	Primary key, always unique
CalHeaderID	Int	N	Foreign key to <b>CalendarHeaders</b> .CalHeaderID, specifying the calendar that this detail applies to
Kind	Int	Y	Always has value 0 (not used by system)
Description	Char(50)	Y	Description of calendar rule
OperationID	Int	Y	Foreign key to <b>Operations</b> .OperationID, specifying the operation that this detail applies to (or 0 for all operations)
FromDate	Int	Y	Used for determining starting date of validity, depending on FromDateKind <sup>1</sup>
ThruDate	Int	Y	Used for determining ending date of validity, depending on ThruDateKind <sup>1</sup>
FromTime	Int	Y	Used for determining starting time of validity (on the starting date of validity), depending on FromTimeKind ^2, 3^
ThruTime	Int	Y	Used for determining ending time of validity (on the ending date of validity), depending on ThruTimeKind ^2, 3^
FromDateKind	Char(2)	Y	Specifies how the system should compute the FromDate date <sup>1</sup>
ThruDateKind	Char(2)	Y	Specifies how the system should compute the ThruDate date <sup>1</sup>
FromTimeKind	Char(2)	Y	Specifies how the system should compute the FromTime time <sup>2</sup>
ThruTimeKind	Char(2)	Y	Specifies how the system should compute the ThruTime time <sup>2</sup>
DailyFromTime	Int	Y	Starting time of validity on any date within FromDate and ThruDate <sup>3</sup>
DailyThruTime	Int	Y	Ending time of validity on any date within FromDate and ThruDate <sup>3</sup>
Sunday	Char(2)	Y	"Y" if valid on Sundays, "N" otherwise
Monday	Char(2)	Y	"Y" if valid on Mondays, "N" otherwise
Tuesday	Char(2)	Y	"Y" if valid on Tuesdays, "N" otherwise
Wednesday	Char(2)	Y	"Y" if valid on Wednesdays, "N" otherwise
Thursday	Char(2)	Y	"Y" if valid on Thursdays, "N" otherwise
Friday	Char(2)	Y	"Y" if valid on Fridays, "N" otherwise
Saturday	Char(2)	Y	"Y" if valid on Saturdays, "N" otherwise
Based	Char(2)	Y	If using relative dates or times, this value indicates what date and/or time from which to base the new date and/or time <sup>3</sup>
Master	Char(2)	Y	"Y" if this rule must be valid in order for calendar to be valid, "N" if calendar can be valid even if this rule is not valid (as long as all rules with Master="Y" are valid, and at least one rule with Master="N" is valid)
OriginalScope	Char(2)	Y	Indicates if this rule originally was a block out rule <sup>4</sup>
OriginalBORule	Int	Y	Currently used to store the original rule number for block outs (this field will not be used in the future)
OrigSunday	Char(2)	Y	Stores the original value of "Sunday" field for block outs (this field will not be used in the future)
OrigMonday	Char(2)	Y	Stores the original value of "Monday" field for block outs (this field will not be used in the future)
OrigTuesday	Char(2)	Y	Stores the original value of "Tuesday" field for block outs (this field will not be used in the future)
OrigWednesday	Char(2)	Y	Stores the original value of "Wednesday" field for block outs (this field will not be used in the future)
OrigThursday	Char(2)	Y	Stores the original value of "Thursday" field for block outs (this field will not be used in the future)
OrigFriday	Char(2)	Y	Stores the original value of "Friday" field for block outs (this field will not be used in the future)
OrigSaturday	Char(2)	Y	Stores the original value of "Saturday" field for block outs (this field will not be used in the future)
Facility	Int	Y	Foreign key to Facility.IdNo. A calendar can be associated to a facility. If non-zero, and the calendar detail is referenced as part of a calendar operation at an ACP belonging to a different facility or to no facility, the calendar is not valid.
Attraction	Int	Y	Foreign key to Attractions.IDNo. A calendar can be associated to an attraction. If non-zero, and the calendar detail is referenced as part of a calendar operation at an ACP functioning as an entrance to a different attraction (or no attraction at all), the calendar detail is not valid.
IgnoreReservationRequirementOnFirstUse	Bit	Y	This only applies when the OriginalScope is "Reservation Required". The option controls whether or not the first use will bypass the reservation check

### Indexes

Name	Kind	Columns	Purpose
PKCalendarDetailsCalDetailID	P	CalDetailID	Primary key.

<sup>1</sup> FromDateKind, ThruDateKind Values

Value	Gateway Constant Name	Description
A	IS_ABSOLUTE	Field described (FromDate or ThruDate) is an actual date in the format YYYYMMDD
R	IS_RELATIVE	Field described (FromDate or ThruDate) is relative to a base date, stored in the format YYMMDD, where YY, MM, and DD are the number of years, months, and days added to the base date
B	IS_BEFORE	Field described (FromDate or ThruDate) is relative before a base date, stored in the format YYMMDD, where YY, MM, and DD are the number of years, months, and days subtracted from the base date
M	IS_RELATIVE_MONTH	Field described (FromDate or ThruDate) is relative to a base date, similar to "R" (format YYMMDD), but using the "relative months" method of calculation

<sup>2</sup> FromTimeKind, ThruTimeKind Values

Value	Gateway Constant Name	Description
A	IS_ABSOLUTE	Field described (FromTime or ThruTime) is an actual time <sup>3</sup>
R	IS_RELATIVE	Field described (FromTime or ThruTime) is relative to a base time <sup>3</sup> , where the Time specified is number of minutes added to the base time.
B	IS_BEFORE	Field described (FromTime or ThruTime) is relative before a base time <sup>3</sup> , where the Time specified is number of minutes subtracted from the base time.

**<sup>3</sup> Note: Only the description field has been changed in order to reflect the changes made to the Time Format.**

Time values are stored as minutes past midnight (12:00/00:00), meaning midnight is represented as 0 and each minute past midnight is added to zero. For example, a time of 3:00 (AM) will be represented as 180 (180 minutes past midnight, i.e., three hours \* 60 minutes in each hour or  $3 * 60 = 180$ ). Likewise, a time of 11:59 PM (23:59) would be stored as 1439 (1439 minutes past midnight, i.e., 23 hours \* 60 minutes in each hour + 59 minutes, or  $(23 * 60) + 59 = 1439$ ).

**<sup>3</sup> Based Values**

Value	Gateway Constant Name	Description
S	RELATIVE_TO_SALE	Relative dates and times are based on the date/time of sale
U	RELATIVE_TO_USE	Relative dates and times are based on the first ticket usage
E	RELATIVE_TO_EVENT	Relative dates and times are based on an event date/time
A	RELATIVE_TO_ARRIVAL	Relative dates and times are based on the order arrival date/time
T	RELATIVE_TO_TKTDATE	Relative dates and times are based on the ticket date
X	RELATIVE_TO_MEMBERSHIP_EXPIRATION	Relative dates and times are based on membership expiration

**<sup>4</sup> OriginalScope Values**

Value	Gateway Constant Name	Description
V	VALID_RULE	Original scope was "Valid Dates"
B	BLOCK_OUT_RULE	Original scope was "Block Out"
R	RESERVATION_REQUIRED_RULE	Original scope was "Reservation Required"

## 4.26 Facility

The **Facility** table contains information about Multipark locations. Each facility contains a set of Access Control Points (see ACPs).

### Columns

Column	Type	Allow Nulls	Description
FacilityID	Int	N	Primary key, always unique. System generated.
IDNo	Int	N	User definable facility ID number. This is the value used to identify a facility throughout the system.
Name	Char(8)	N	Name of the facility.
Descr	Char(40)	N	Description of the facility.
Capacity	Int	N	Capacity of the Facility. <i>Not currently used by system.</i>
VelocityTimeLimit	Int	Y	The time period after which a customer will be allowed to reenter this facility.

### Indexes

Name	Kind	Columns	Purpose
PKFacilityFacilityID	P	FacilityID	Primary Key
PKFacilityIDNo		IDNo	Unique constraint on Facility.IDNo

## 4.27 FacilityAccess

The FacilityAccess table is created automatically by TCON32, if the "Create FacilityAccess" option is selected. FacilityAccess records are created for every valid usage date for access code definitions with a BankHeaderID or CallHeaderID defined.

### Columns

Column	Type	Allow Nulls	Description
FacAccessID	Int	N	Primary key, always unique
AccessCode	Int	Y	Foreign key to AccessCodes.AccessCode
FacilityID	Int	Y	Foreign key to Facility.IdNo
NdaysFromTicket	Int	Y	Specifies a date (as an integer) that a ticket can be used

### Indexes

Name	Kind	Columns	Purpose
PKFacilityAccessFacAccessID	P	FacAccessID	Primary Key.

## 4.28 FacilityCounts

The FacilityCounts table contains a record of the total Entries, Reentries and Exits for each Facility.

### Columns

Column	Type	Allow Nulls	Description
FacilityID	Int	N	The Facility that the counts were generated from. (Primary key, always unique)
Date	DateTime	N	The working date for the counts
EntryCounts	Int	N	The number of entries that have occurred
ReentryCounts	Int	N	The number of reentries that have occurred
CrossoverCounts	Int	N	The number of crossovers that have occurred
EntryClicks	Int	N	The number of entry clicks recorded if a Turnstile click mechanism is installed
ReentryClicks	Int	N	The number of reentry clicks recorded if a Turnstile click mechanism is installed
ExitClicks	Int	N	The number of exit clicks recorded if a Turnstile click mechanism is installed

### Indexes

Name	Kind	Columns	Purpose
PKFacilityCounts	P	FacilityID, Date	Primary Key

## 4.29 FkeyTasks

The FkeyTasks table contains the function key definitions for the Access Control application (ACS2), which allows the user to assign system defined functions to the terminal function keys.

### Columns

Column	Type	Allow Nulls	Description
FkeyTaskID	Int	N	Primary key, always unique. System generated.
Fkey	Int	N	The terminal function key number the function is assigned to.
Name	Char(7)	Y	The description of the function key.
Task	Int	Y	System defined tasks for the Fkey. <sup>1</sup>
Data	NVarChar(180)	Y	Contains any data associated with the Task.
FkeyLevel	Int	N	Function key level to support multiple functions to a Fkey
Descr	Char(40)	Y	Long description

### Indexes

Name	Kind	Columns	Purpose
PKFKeyTasksFkeyTaskID	P	FkeyTaskID	Primary Key.

### <sup>1</sup> Task Values

Value	Gateway Constant Name	Descriptions
1	LOGOFF_TASK	Log current agent off of terminal.
2	SCAN_TASK	Enter a pre-defined scan, or allow entry from keyboard.
3	PAGE_TASK	Page Tlead station.
4	ADMIT_TASK	Admit ticket holder (record usage) if last scan was invalid.
5	HELP_TASK	Display help message.
6	TIME_TASK	Display current date/time.
7	MESSAGES_TASK	View messages from Tlead.
8	DATE_TASK	Set the date/time stamp for scanning.
9	RELOCK_TASK	Relock turnstile.
10	MENU_TASK	Switch to another menu.
11	EXIT_MODE_TASK	Turnstile: Allow exit only. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
12	ENTRY_MODE_TASK	Turnstile: Allow enter only. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
13	PAID_MODE_TASK	Turnstile: Allow PAID enter and exit. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
14	FREE_MODE_TASK	Turnstile: Allow FREE entry/exit. This function is obsolete: it has been replaced by ACP_MODE_TASK.
15	SHOW_MODE_TASK	Turnstile: Display the current ACP mode last set by invoking the ACP_MODE_TASK.
16	GETVER_TASK	Turnstile: Display version.
17	RESTART_TASK	Turnstile: Restart.
18	RECORD_TASK	Turnstile: Record Entry.
19	TEMPCLOSE_TASK	Turnstile: Close temporarily.
20	UNLOCK_TASK	Turnstile: Unlock mode.
21	GROUP_EXIT_TASK	Group Exit.
22	CLOSE_TASK	Close station.
23	LOCKBAR_TASK	Turnstile: Lock turnbar once.
24	UNLOCKBAR_TASK	Turnstile: Unlock turnbar once.
25	REPLACE_OPER_TASK	Replace current operation.
26	CLEAR_OPER_TASK	Clear all active operations.
27	ADD_OPER_TASK	Add a single operation.
28	REMOVE_OPER_TASK	Remove a single operation.
29	SHOW_OPER_TASK	Show active operations.
30	ACP_MODE_TASK	Set ACP Mode, or show menu
31	MANAGE_DEVICES_TASK	Manage devices in ACS32
32	EXIT_TASK	Exit application in ACS32
33	MANAGE_MODULES_TASK	Manage AX1180 modules in ACS32
34	AUTOSCAN_TASK	AutoScanning in ACS32
35	SHOW_CONSOLE_TASK	Show console in ACS32
36	TICKET_LOOKUP_TASK	Ticket lookup in ACS32
37	NEXT_TICKET_LOOKUP_TASK	Lookup next ticket scanned in ACS32
38	START_SELF_SERVICE_TASK	Start ACP Self-Service Mode
39	STOP_SELF_SERVICE_TASK	Stop ACP Self-Service Mode
40	TOGGLE_BIOMETRICS_TASK	Stop using biometrics until restart
41	TOGGLE_LOGGING_TASK	Toggle all logging
42	EXIT_GALAXY_TASK	Exit the Galaxy Point of Sale application (used only when ACS32 is running within Galaxy)
43	CAPTURE_BIOMETRIC_TASK	Performs Registration of Biometric data
44	RELOCK_ALL_TASK	Relock the turnbar for all pending guests. This function removes all pending rotations on the turnbar.
45	SHOW_BIOMETRICS_STATE_TASK	Show whether biometrics are currently enabled or disabled.
46	ENABLE_BIOMETRICS_TASK	Enable biometric activities.
47	DISABLE_BIOMETRICS_TASK	Disable biometric activities.

48	SHOW_BIOMETRICS_RATE_TASK SET_BIOMETRICS_RATE_TASK	Show the current rate at which biometric identifications are being performed. Set the rate at which to perform biometric identifications.
50	SHOW_LOGGING_TASK	Show whether the logging function has been invoked to enable diagnostic logging.
51	ENABLE_LOGGING_TASK	Enable all diagnostic logging. All logging categories are enabled and ACS32 logs everything directly to the log file. (Logging is automatically disabled when a user logs off or exits ACS32).
52	DISABLE_LOGGING_TASK	Disable diagnostic logging. All log categories are turned off.
53	EXECUTE_GALAXY_FUNCTION_TASK	Execute galaxy function in ACS@POS
54	CHANGE_ACP_TASK	Change ACP to a new ACP
55	ENABLE_VEHICLE_DETECTOR_TASK	Enables vehicle detector
56	DISABLE_VEHICLE_DETECTOR_TASK	Disables vehicle detector
57	TOGGLE_VEHICLE_DETECTOR_TASK	Toggles vehicle detector
58	TOGGLE_BYPASS_VEHICLE_DETECTOR_TASK	Bypass vehicle detector
59	REINITIALIZE_VEHICLE_DETECTOR_TASK	Reinitialize vehicle detector
60	TOGGLE_SURVEY_PROMPT_TASK	Toggle survey prompt
61	TOGGLE_VEHICLE_SURVEY_PROMPT_TASK	Toggle vehicle survey prompt
62	RETAKE_PHOTO	Retake photo
63	TOGGLE_SHOW_SECURE_DATA	Toggle whether or not to obscure secure data.
64	SHOW_USER_BROWSER_TASK	Shows the User Browser.
65	MODIFY_TICKET_DATE	Modify the ticket date of the last scanned ticket in ACS
66	QUICK_ORDER_VALIDATION_TASK	Open the Quick Order Validate wizard in ACS@POS
67	SEND_TURNSTILE_MESSAGE_TASK	Currently not implemented
68	OVERRIDE_REENTRY_TASK	Overrides scan like ADMIT_TASK, but overrides as a reentry so the guest is not charged.
69	LOGON_TASK	Logon agent to terminal for remote ACP

### 4.30 HSAccessCodes

Stores Handshake specific configuration data for a given access code.

#### Columns

Column	Type	Allows Nulls	Description
HSAccessCodeID	Int	N	Primary key, always unique. System generated
AccessCodeID	Int	N	Access code ID to associate this record to
Encoding <sup>1</sup>	Int	Y	Defines whether or not a barcode is used
Visible <sup>2</sup>	Int	Y	If a barcode is being used, defines if it's visible or infrared
BarcodeType <sup>3</sup>	Int	Y	If using a barcode, the type of barcode
RFChip <sup>4</sup>	Int	Y	If not using barcode, the type of RF chip
UsePurge	Bit	Y	If 1, tells Handshake to purge the ticket from its database based on the date of sale and the PurgeYears, PurgeMonths and PurgeDays settings
PurgeYears	Int	Y	Number of years Handshake waits after the date of sale to purge tickets with this access code
PurgeMonths	Int	Y	Number of months Handshake waits after the date of sale to purge tickets with this access code
PurgeDays	Int	Y	Number of days Handshake waits after the date of sale to purge tickets with this access code
SendPackages	Bit	Y	Defaults to NULL/0. If 1, and a package is being processed, it will be encoded to RFID if RFID encoding is enabled on the HSAccessCode. Eventually this will also control whether Handshake Ticket Sender in Galaxy sends a package referencing this HSAccessCode to Handshake. Currently only individual media packages are sent to Handshake. This option allows you to use the same Access Code for a Package and a Handshake Ticket without causing the Package to be encoded to a SKIDATA RFID card.

#### Indexes

Name	Kind	Columns	Purpose
PKHSAccessCodesID	P	HSAccessCodesID	Primary key

#### <sup>1</sup> Encoding Values

Value	Gateway Constant	Description
0		Using a barcode
1		Not using a barcode

#### <sup>2</sup> Visible Values

Value	Gateway Constant	Description
0		Infrared barcode
1		Visible barcode

#### <sup>3</sup> BarcodeType Values

Value	Gateway Constant	Description
0		Standard 2 of 5
1		Interleaved 2 of 5
2		UPC A
3		UPC E
4		EAN
5		Code 39
6		Code 128
7		Codabar
8		Code 93
9		MSI
10		Plessey

#### <sup>4</sup> RFChip Values

Value	Gateway Constant	Description
8		RF-Chip V4150 Keycard
11		RF-Chip V4150 Swatch Access
13		Gem C210
14		GemC240
15		Gem C285
16		Gem C200
18		Legic

### 4.31 LockOuts

The tables stores Galaxy's Lockout information.

#### Columns

Column	Type	Allow Nulls	Description
LockoutID	Int	N	Primary key, always unique
AccessCode	Int	Y	
Serial	Char(20)	Y	
HiSerial	Char(20)	Y	
LockOutMsg	Char(80)	Y	
SoundFile	Char(12)	Y	
AllowAccess	Bit	Y	
PageLead	Bit	Y	
LongReason	Char(255)	Y	
FromVisualID	VarChar(40)	Y	The first visual ID to be locked out if using a visual id range. In this case, this will be parsed and the result will be used to fill the AccessCode and Serial fields.
ThruVisualID	VarChar(40)	Y	The last visual ID to be locked out if using a visual id range. In this case, this will be parsed and the result will be used to fill the AccessCode and HiSerial fields.
ExpirationDate	Datetime	Y	Date after which the lockout record may be deleted via MWS, Access Control > Lockouts > Remove Expired menu item. After removing the expired lockouts the Lockout database should be sent to the ACPs. The goal is to keep the Lockout database as smaller as possible at the ACPs that may have space constraints.

#### Indexes

Name	Kind	Columns	Purpose
PKLockoutsLockoutID	P	LockoutID	Primary Key.
IXLockoutsAccessCodeSerial		AccessCode, Serial, HiSerial	Index to speedup lockout record lookup using AccessCode, Serial, and HiSerial
IXLockoutsFromThruVisualID		FromVisualID, ThruVisualID	Index to speedup lockout record lookup using FromVisualID and ThruVisualID

## 4.32 Media

The **Media** table contains information about barcodes used by the system. Galaxy and ACS2 use a local version of this table (MEDIA.DAT).

### Columns

Column	Type	Allow Nulls	Description
MediaID	Int	N	Primary key, always unique.
Name	Char(20)	Y	Name of the media definition.
LoPrefix	Char(20)	Y	Lowest barcode prefix to be recognized by this media definition.
HiPrefix	Char(20)	Y	Highest barcode prefix to be recognized by this media definition.
Len	Int	Y	Length of barcodes to be recognized by this media definition.
IDStart	Int	Y	Starting position of the serial number within the barcode.
IDLen	Int	Y	Length of the serial number.
NodeStart	Int	Y	Starting position of the node number within the barcode.
NodeLen	Int	Y	Length of the node number.
AccessCodeStart	Int	Y	Starting position of the access code within the barcode.
AccessCodeLen	Int	Y	Length of the access code.
AccessCode	Int	Y	Foreign key to AccessCodes.AccessCode, specifying the access code to be used if the access code is not in the barcode, or 0 otherwise.
DateStart	Int	Y	Starting position of the barcode date.
DateLen	Int	Y	Always 0. This field is not currently used by the system.
DateFormat	Int	Y	Specifies the format of the barcode date, if used. <sup>1</sup>
Kind	Int	Y	Specifies the kind of media described by this definition. <sup>2</sup>
QtyStart	Int	Y	Starting position of the quantity within the barcode, if used.
QtyLen	Int	Y	Length of the quantity section.
DateKind	Int	Y	If a date is encoded in the barcode, this field specifies how the system should determine validity from the barcode date. <sup>3</sup>
MatchAccess	Bit	Y	If selected, the AccessCodeStart, AccessCodeLen, and AccessCode are used to match the media definition.
Encoding	Int	Y	Specifies encoding scheme used to scramble sequence number in the barcode. <sup>4</sup>
CompanyStart	Int	Y	Starting position of the company number within the barcode.
CompanyLen	Int	Y	Length of the company number.
Mask	Char(30)	Y	Mask for Visual ID.
ValueStart	Int	Y	Starting position of the value.
ValueLen	Int	Y	Length of the value.
ValueMulti	Int	Y	Multiplier value.
MissingPrompt	Char(20)	Y	Prompt for another barcode.
MissingLen	Int	Y	Length of other barcode.
IsPrimary	Bit	Y	Specifies if this is the primary barcode.
AccessMask	Char(30)	Y	Mask to prefix access codes (xxxx@ACCESSyyyy).
SequenceStart	Int	Y	Starting position of the sequence number within the barcode.
SequenceLen	Int	Y	Length of the sequence number.
CheckDigitStart	Int	Y	Starting position of the check digit within the barcode, if used.
CheckDigitLen	Int	Y	Length of the check digit section.
CheckDigitMethod	Int	Y	Method used for computing check digit validity, if used. <sup>5</sup>
PreFetchMode	Int	Y	Mode to determine if ACS needs to get the record before validating. <sup>8</sup>
MediaType	Int	Y	Where the scan was read from <sup>6</sup>
ConstructVisualID	Bit	N	If true, the Visual ID is built using the information provided in MediaVisualIDDetails
LockoutLookupMethod	Int	Y	Used to indicate how to lookup a lockout record <sup>7</sup>
PLUStart	Int	Y	Start of the PLU
PLULength	Int	Y	Length of the PLU
ReplaceAccessCode	bit	Y	If true, replace Access Code in Visual ID with Access Code defined in Media row when validating
ExternalID	Char(20)	Y	External ID
PreFetchPluginID	Int	Y	Foreign key to Plugins.PluginID specifying the plugin used for Lookups.

### Indexes

Name	Kind	Columns	Purpose
PKMediaMediaID	P	MediaID	Primary Key.

<sup>1</sup> DateFormat Values

Value	Gateway Constant Name	Description
1	DATE__YYMMDD	Barcode date is in "YYMMDD" format (May 17 <sup>th</sup> , 2003 would be encoded as "030517").
2	DATE_MMDD	Barcode date is in "MMDD" format (May 17 <sup>th</sup> would be encoded as "0517").
3	DATE__YYYYMMDD	Barcode date is in "YYYYMMDD" format (May 17th, 2003 would be encoded as "20030517").
4	DATE__YYMM	Barcode date is in "YYMM" format (May, 2003 would be encoded as "0305").
5	DATE__YYMMTT	Barcode date is in "YYMMTT" format, where "TT" is the number of months the barcode is valid (May and June of 2003 would be encoded as "030502").
6	DATE__YMMDD	Barcode date is in "YMMDD" format (May 17 <sup>th</sup> , 2003 would be encoded as "30517").

<sup>2</sup> Kind Values

Value	Gateway Constant Name	Description
0	TICKET_MEDIA_DEF	Ticket barcodes.
1	DISCOUNT_MEDIA_DEF	Discount barcodes.
2	ITEM_MEDIA_DEF	Item barcodes.
3	VOUCHER_MEDIA_DEF	Voucher barcodes.
4	FUNCTION_MEDIA_DEF	Function barcodes.
5	PASS_MEDIA_DEF	Pass media definition
6	LOGON_CARD_MEDIA_DEF	Logon card media definition
7	LOYALTY_MEDIA_DEF	Loyalty media definition
8	TRANSACTION_LOOKUP_MEDIA_DEF	Transaction Lookup barcode media definition
9	SERIALIZED_STATEMENT	Serialized statement media def
10	CONTACT_MEDIA_DEF	Contact identifier barcode

**3 DateKind Values**

Value	Gateway Constant Name	Description
0	DATE_KIND_THROUGH	Barcode is valid before or on the barcode date.
1	DATE_KIND_ON_AFTER	Barcode is valid on or after the barcode date.
2	DATE_KIND_ON_ONLY	Barcode is valid on the barcode date only.

**4 Encoding Values**

Value	Gateway Constant Name	Description
0	ENCODE_NONE	The visual id is not encrypted
1	ENCODE_SERIAL	The serial number is encrypted
2	ENCODE_SEQUENCE	Sequence number is encrypted in the visual id
3	ENCODE_SEQUENCE_8TO11	The sequence number is encrypted in the visual ID. An 8 digit sequence is converted to an 11 digit random number.
4	ENCODE_USING_PLUGIN	The visual ID is encoded using an ID encoder plugin.
5	DECODE_USING_PLUGIN	The visual ID is decoded using an ID encoder plugin.

**5 CheckDigitMethod Values**

Value	Description
0	System uses "Method 1" of computing check digit validity.
1	System uses "Method 2" of computing check digit validity.

**6 MediaType Values**

Value	Gateway Constant Name	Description
0	ANY_MEDIA_TYPE	The source of the scan is not known (default)
1	TRACK1_MEDIA_TYPE	The scan is read from track 1 of the ticket (or credit card type of ticket)
2	TRACK2_MEDIA_TYPE	The scan is read from track 2 of the ticket (or credit card type of ticket)
3	TRACK3_MEDIA_TYPE	The scan is read from track 3 of the ticket (or credit card type of ticket)
4	OMRON_BAR_CODE_MEDIA_TYPE	The scan is read from the barcode reader attached to an Omron turnstile
5	OMRON_NON_CONTACT_MEDIA_TYPE	The scan is read from a non-contact interface attached to an Omron turnstile
6	OMRON_JR_TICKET_MEDIA_TYPE	The scan is read from a JR ticket reader attached to an Omron turnstile

**7 LockoutLookupMethod Values**

Value	Gateway Constant Name	Description
0	LOCKOUT_LOOKUP_BY_ACCESS_CODE	Lookup lockout by AccessCode & Serial
1	LOCKOUT_LOOKUP_BY_VISUAL_ID	Lookup lockout by VisualID

**8 PrefetchMethod Values**

Value	Gateway Constant Name	Description
0	PREFETCH_NONE	"Access Code Without Entitlement Add-On (Standard)". No prefetch, the access code is used to determine what table is used for the lookup. Not used for entitlement add-ons. The media definition must either specify the location of the access code in the visual ID or specify the access code directly in the AccessCode field.
1	PREFETCH_TICKET	"Ticket". Get Ticket record before validating. An access code is not required to be configured on the media definition.
2	PREFETCH_PASS	"Pass". Get Pass record before validating. An access code is not required to be configured on the media definition.
3	PREFETCH_SUPERTICKET	"SuperTicket". Get SuperTicket record before validating. An access code is not required to be configured on the media definition.
4	PREFETCH_PLUGIN	"Plugin Name". Get information from a plugin before validating. An access code is not required to be configured on the media definition.
5	PREFETCH_BY_ACCESS_CODE	"Access Code With Entitlement Add-On". Use the access code to determine what table to use for the lookup. A media def that has a prefetch mode of 5 can be used on a ticket, pass, and debit at the same time. This option will allow for the validation of the super ticket chain generated by entitlement add-ons. The media definition must either specify the location of the access code in the visual ID or specify the access code directly in the AccessCode field.

### 4.33 MediaIDDetails

This table defines how to construct a media number (credit card / ticket number) from a scan. This table basically gives information how to selectively pickout characters from a scan and construct an ID number. This table will link to records in the Media table (for constructing ticket Visual IDs) and Bin table (for constructing credit card numbers).

#### Columns

Column	Type	Allow Nulls	Description
MediaIDDetailID	Int	N	Primary key, obtained from GatewayCounters
AuxID	Int	Y	Foreign key to Media.MediaID or Bins.BinID
Kind	Int	Y	Indicates FK link (0 for Media; 1 for Bins) <sup>2</sup>
Sequence	Int	Y	Sequence to the entry (currently 1 - 10)
ConstructIDField	Int	Y	Field name from standard magstripe track field names <sup>1</sup>
ConstructIDStart	Int	Y	Start position of data for Field
ConstructIDLength	Int	Y	Length of data for Field

#### Indexes

Name	Kind	Columns	Purpose
PKMediaIDDetailID	P	MediaVisualIDDetailID	Primary Key.
IXMediaIDDetailsAuxID	F	AuxID	Can be used to query for all fields relating to a single media or bin record

<sup>1</sup> ConstructVisualIDField Values

Value	Gateway Constant Name	Description
0	FM_RAW_DATA	The entire original contents of the scan/swipe.
1	FM_ACCOUNT_NUMBER_FIELD	The PAN (Primary Account Number) field in the track data (Maximum of 19 characters)
2	FM_NAME_FIELD	The Name field in the track data (Maximum of 26 characters, only available on Track 1)
3	FM_ADDITIONAL_NAME_FIELD	The Additional Data field in the track data (includes ExpirationDate and ServiceCode_)
4	FM_DISCRETIONARY_DATA_FIELD	The Discretionary Data field in the track data

<sup>2</sup> Kind Values

Value	Gateway Constant Name	Description
0	MEDIA_DEF_DETAIL_KIND	AuxID references a record in the Media table (AuxID -> MediaID)
1	BIN_DEF_DETAIL_KIND	AuxID references a record in the Bins table (AuxID -> BinID)

#### 4.34 MemberUsage

The MemberUsage table contains information that relates to the usage/validation of a membership.

Column	Type	Allow Nulls	Description
MemberUsageID	Int	N	Primary key for this table. Value is from GatewayCounters and is sequentially incremented for each row.
UsageID	Int	N	FK link to Usage.UsageID. This link connects this record to the usage record that it is associated with
AdultQty	Int	Y	The number of adults admitted on this validation
ChildQty	Int	Y	The number of children admitted on this validation
ScannedVisualID	Varchar(40)	Y	The visual ID that was scanned for the membership. If a superticket visual ID is scanned, this will contain that visual ID. The visual ID on the usage will be replaced by the visual ID of the referenced ticket or pass.
GuestQty	Int	Y	Stores the number of guests admitted through an Admission Control Point with a Joint Membership Pass.
JointMemberID	Int	Y	FK to JointMembers.JointMemberID. This indicates the JointMember that the usage was generated for.

#### Indexes

Name	Kind	Columns	Purpose
PKMemberUsageID	P	MemberUsageID	Primary Key.

### 4.35 Operations

The **Operations** table contains access control "operation" data. Each facility can have multiple modes of operation.

#### Columns

Column	Type	Allow Nulls	Description
OperationUniqueID	Int	N	Primary key, always unique. System generated.
OperationID	Int	N	User definable operation ID number. This is the value used to identify an operation throughout the system.
Description	Char(50)	Y	The operation description.
Facility	Int	Y	Foreign key to Facility.IdNo, specifying the facility this operation applies to which this operation applies.
Points	Integer	Y	The number of points charged for point redemption of a stored value card when an ACP is in this operation.
Attraction	Integer	Y	Foreign key to Attractions.IDNo

#### Indexes

Name	Kind	Columns	Purpose
PKOperationsOperationUniqueID	P	OperationUniqueID	Primary key
PKOperationsOperationID		OperationID	Unique constraint on Operation.OperationID

### 4.36 PendingUsage

PendingUsage table stores the Usage records that are used for further processing by other applications. PendingUsage table definition is similar to the definition of Usage table except a few new columns are added to the PendingUsage table. Currently data in the PendingUsage table is inserted by the insert trigger on the Usage table.

#### Columns

Column	Type	Allow Nulls	Description
PendingUsageID	Int	N	Primary key, always unique. This is an identity column and values in this column are generated by SQL server automatically.
UseTime	Datetime	N	The time at which activity occurred at the ACP.
VisualID	Varchar(40)	N	The barcode as read from the scanner.
AccessCode	Int	N	The AccessCode extracted from the barcode scan which generated the Usage record.
IDNo	Char(40)	N	The barcode as read from the scanner, same as VisualID.
Code	Int	N	The type of activity Valid values for this column are shown in the <i>Code Values</i> table of "Usage" table (please refer to table definition of Usage table)
ACP	Int	N	The ID number of the Access Control Point at which the activity took place.
Status	Int	N	The result of the activity, as defined in the <i>Status Values</i> table of "Usage" table (please refer to table definition of Usage table)
Qty	Int	N	The number of guests admitted or value redeemed.
UseNo	Int	N	The number of time the ticket or pass was used.
Operator	Int	N	Foreign key into GXUsers.UserID, referencing the user logged on at the time the Usage record was created. This value is 0 if nobody was logged-on
EntryMethod	Int	N	A number representing the method used to enter the barcode into the system which in-turn generated the Usage record. Possible values of EntryMethod are defined in the <i>EntryMethod Values</i> table of "Usage" table (please refer to table definition of Usage table)
SerialNo	Char(20)	N	The ID portion of the ticket which was scanned to produce the Usage record
Override	Int	N	If the operator overrides an invalid ticket and lets the guest into the park, this field stores the original reason the ticket was rejected <sup>1</sup>
UsageCondition	Int	Y	A number representing the condition of TCon32 when the ticket was scanned. Possible values of UsageCondition are defined in the <i>UsageCondition Values</i> table of "Usage" table (please refer to table definition of Usage table)
OriginalStatus	Int	Y	A number representing the value of the Usage Status, before running the Offline Recategorization program, "ResetUse".
BankNo	Int	Y	The bank detail number validated by ACS2. In other words, the bank detail that was valid and allowed the admission. This bank number is sent to TCON32 which uses it to determine the correct ticket bank detail to remove uses from.
BiometricStatus	Int	Y	Status of Biometric activity performed during this usage. Possible values of BiomtricStatus are defined in the <i>Biometric Values</i> table of "Usage" table (please refer to table definition of Usage table)
OperationID	Int	Y	Operation that was used
ProcessCode <sup>1</sup>	Int	Y	Value in this column indicates what action needs to be performed on this usage record. Possible values of ProcessCode are defined in the <i>ProcessCode Values</i> table below.
ProcessStatus <sup>2</sup>	Int	Y	Value in this column indicates the current status of the pending usage record. Possible values of ProcessStatus are defined in the <i>ProcessStatus Values</i> table below.
GxEventLogID	Int	Y	FK reference to GxEventLog.GxEventLogID column
ScannedVisualID	Varchar(40)	Y	The visual ID of the barcode that is scanned. This may differ from the visual ID if the scanned barcode is from a SuperTicket record.
AttractionID	Int	Y	The Attraction that was configured on the ACP where this usage occurred. Foreign key to Attractions.IDNo.
OriginalUsageID	Int	Y	The original usage id that this usage was based off of. Used for Reversals and Voids of usages.
FacilityID	Int	Y	Foreign key to Facility.IDNo. ACPs can now function as an exit of a facility. If zero, the facility that the usage is applied to can be obtained via Usage.ACPL. If non-zero, the usage applies to the indicated facility.
EntitlementCharged	Bit	Y	If 1, the entitlement has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION.
BankCharged	Bit	Y	If 1, the bank has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION, and the same bank was used for both admissions.
NodeNo	Int	Y	The node number this activity occurred at.
TransNo	Int	Y	The POS transaction number this activity occurred in (if available).
PLU	nVarChar	Y	The PLU of the scanned item (if available).
ExternalUsage	bit	Y	Indicates if the usage originated from an external validation.
ACSRReservationConfirmationNumber	nvarchar(50)	Y	The confirmation number of the reservation used during validation.
ProcessingError	Int	Y	<p>If not null and not zero, indicates that a validation or processing error occurred while this usage was being examined by the Usage Processor component in Direct-to-Database ACS32/Acces</p> <p>Many of these errors can only occur if a scan occurs while the database is offline, since the errors would normally result in a ticket not being valid for admission.</p> <p>The purpose of this column is to report usage processing errors that are related to offline scanning, and to configuration errors, not invalid scans or operator overrides:</p> <ul style="list-style-type: none"> <li>- Most tests are not performed for invalid scans. The Usage.Status column can be used to report invalid scans.</li> <li>- Most tests are not performed when invalid validation results are overridden by operators. Overrides can be found by examining Usage.Override. The conditions under which each test is performed are noted in the associated values table below.</li> </ul> <p>The Exceptions Live Access report displays usages with non-null processing errors. These can be compared to overridden scans in the same report.</p> <p><b>NOTE:</b> Only one ProcessingError value is stored in the Usage table. Checks are performed in the order below. If any check fails, additional checks are not performed.</p> <ol style="list-style-type: none"> <li>1. PE_INVALID_ACP</li> <li>2. PE_INVALID_MEDIA_DEF</li> <li>3. PE_RFIDMAP_NOT_FOUND</li> <li>4. PE_INVALID_ACCESS_CODE</li> <li>5. PE_NO_UPDATE_TABLE</li> <li>6. PE_TICKET_NOT_FOUND, PE_PASS_NOT_FOUND, PE_DEBIT_CARD_NOT_FOUND</li> </ol>

		7. PE_INVALID_SUPER_TICKET_NEXT_ID 8. Status errors
Possible values are documented in the Usage table section.		

**Indexes**

Name	Kind	Columns	Purpose
PKPendingUsageID	P	PendingUsageID	Primary Key.
IXPenUsageProcessCodeStatus		ProcessCode, ProcessStatus	Index to improve querying PendingUsage records from ProcessUsage app

**<sup>1</sup> ProcessCode Values**

Value	Gateway Constant Name	Description
0	RE_MERGE_PROCESS_CODE	Pending usage record is processed by the REMerge app
1	SW_SERVER_PROCESS_CODE	Pending usage record is processed by the SWServer app
2	INVOICE_BY_USE_TIME_PROCESS_CODE	Pending usage record is processed by "Invoice by use time" process in ProcessUsage app

**<sup>2</sup> ProcessStatus Values**

Value	Gateway Constant Name	Description
0	NEW_PENDING_USAGE_RECORD_PROCESS_STATUS	New pending usage record, waiting to be processed
1	REJECTED_PENDING_USAGE_RECORD_PROCESS_STATUS	Pending usage record is rejected

### 4.37 Reentry

This table is used to store ticket numbers that have been scanned per facility for a given working day. Each ticket scanned will be recorded in this table once (per facility) and used to determine reentry and crossover status. The table is truncated once per day as part of the TCON32 end of day process. If this table does not exist or is missing columns, TCON32 will automatically adjust the structure on startup. Previous versions (pre-1.7) of TCON32 supported a locally maintained Paradox reentry table. This is no longer the case, all reentry records are now stored in this reentry DBMS table.

#### Columns

Column	Type	Allow Nulls	Description
AccessCode	Int	N	Foreign key to AccessCodes.AccessCode, specifying the ticket access code.
SerialNo	Char(20)	N	Serial portion of the ticket number.
Facility	Int	N	Foreign key to Facility.IdNo, specifying the facility number where the scan occurred.
EntryTime	DateTime	Y	Date/Time when the scan was recorded.
Qty	Int	Y	Number admitted with this scan.
ACP	Int	Y	Foreign key to ACPs.AcpID, specifying the Access Control Point where the scan occurred.
VisualID	char(40)	Y	Scan code or barcode as read from scanner; combination of AccessCode and ID.
BankNo	Int	Y	Bank Number used for validation
Override	Int	Y	The value from Usage.Override. If this is non-zero the ticket was overridden. This is used by the "Once In Always In" feature in ACS32.
ScannedVisualID	varchar(40)	Y	The actual Visual ID that was scanned instead of the Visual ID of the entitlement associated with the scan. This will be the same as VisualID except when certain types of SuperTickets are scanned. Analogous to Usage.ScannedVisualID.
AdultQty	Int	Y	The number of adults admitted with a joint member pass.
ChildQty	Int	Y	The number of children admitted with a joint member pass.
GuestQty	Int	Y	The number of guests admitted with a joint member pass.
JointMemberID	Int	Y	FK to JointMembers.JointMemberID. This is used to whether a Joint Member entered a facility earlier in the same business day.
Status	Int	Y	The status of the scan, obtained from Usage.Status.
Attraction	Int	Y	Foreign key to Attractions.IDNo. If non-zero, the reentry row was generated by an ACP associated to an attraction.

#### Indexes

Name	Kind	Columns	Purpose
PKReentryReentryID	PK	ReentryID	Primary key, always unique. Sequential counter value is either from GatewayCounters or a SQL sequence, depending on the database configuration.
IXReentryAccessSerialFacility		AccessCode + SerialNo + Facility	This index is used when TCON32 is configured to lookup reentry records by AccessCode + SerialNo. If configured to lookup reentry by VisualID, this index is not used.
IXReentryVisualID		VisualID	This index is used when TCON32 is configured to lookup reentry records by VisualID. If configured to lookup reentry by AccessCode + SerialNo, this index is not used.
IXReentryEntryTime		EntryTime	Used by TCon32
IXReentryLastUpdate	P	LastUpdate	Used in TCon32's End of Day process for deleting Reentry table.
IXReentryFacility		Facility	Used by TCon32
IXReentrySerialNo		SerialNo	Used by TCon32
IXReentryScannedVisualID		ScannedVisualID	Speed up lookups on ScannedVisualID column
IXReentryJointMemberID		JointMemberID	The GTSGetJointMembersByPassNo stored procedure performs a left outer join on JointMembers.JointMemberID and Reentry.JointMemberID to determine whether a Joint Member was admitted earlier the same day. This index is not necessary in installations that do not use Joint Membership passes.

**4.38 RFCS**

Contains SQL version of RFCS Items, which match Galaxy items with the corresponding items defined in the ERG system. RFCS Items need to be defined only if using ERG's GAK smart card reader. This table is populated by DBSync, but is not otherwise used by the system.

**Columns**

Column	Type	Allow Nulls	Description
RFCSID	Int	N	Primary key, always unique
PLU	nchar(20)	N	Foreign key to Items.PLU, mapping Gateway item to link with ERG
RouteID	Int	Y	Identifies "Route ID" defined in ERG data for corresponding item
TripID	Int	Y	Identifies "Trip ID" defined in ERG data for corresponding item
PassengerType	Int	Y	Identifies "Passenger Type" defined in ERG data for the item <sup>1</sup>
VehicleType	Int	Y	Identifies "Vehicle Type" defined in ERG data for the item <sup>2</sup>
Fare	Money	Y	Not used (always 0.00)
AdditionalFare	Money	Y	Not used (always 0.00)
NSFFlag	Boolean	Y	True if fare is to be deducted even if the value on the smart card is not sufficient to pay for the selection
ProductTypeID	Int	Y	Identifies "Product Type ID" defined in ERG data for the item
ProductIssuerID	Int	Y	Identifies "Product Issuer ID" defined in ERG data for the item. For ngORCA, this is the OperatorID
NGOrcaLine	nchar(20)	Y	The ngORCA "Line" number

**Indexes**

Name	Kind	Columns	Purpose
PKRFCSID	P	RFCSID	Primary Key.

<sup>1</sup> PassengerType Values

Value	Description
0	Invalid
1	Adult
2	Youth
3	Senior
4	Disabled
5	Low Income
255	Unspecified (read from card)

<sup>2</sup> VehicleType Values

Value	Gateway Constant Name	Description
0	None	No Vehicle
1	Bicycle	Bicycle
2	Motorcycle	Motorcycle
3	OversizedMotorcycle	Oversized Motorcycle
4	Under20	Vehicle under 20'
5	OverHeight	Vehicle under 20' Overheight
6	Carpool	Carpool
7	Vanpool	Vanpool
8	Other	Other
254	Any	Any
255	VehicleNotSpecified	Not Specified

### 4.39 RFCSUsage

Contains Usage Data specific to the Regional Fare Card System.

Usage.VisualID can be used as a foreign key to this database, formatted as PurseTxnRef1 + PurseTxnRef2 + ProductTxnRef1 + ProductTxnRef2, each left-justified to 10 digits and padded with zeroes (e.g. 00000000010000000020000000003000000004).

Similarly, PurseTxnRef1 + PurseTxnRef2 + ProductTxnRef1 + ProductTxnRef2 can be concatenated to form a foreign key to Usage.VisualID.

Column	Type	Description
RFCSUsageID	Int	Primary Key, Unique
Status	Int	Status of Transaction <sup>1</sup>
CardSerialNumber	BigInt	Serial number of smartcard used in transaction
TransType	Int	Transaction Type <sup>2</sup>
PurseAmountDeducted	Money	Value deducted from purse on card (0 if TransType <> ttPurse)
PurseRemainingValue	Money	Value remaining on purse (0 if TransType <> ttPurse)
TripDiscountAmount	Money	Discount applied to transaction
PurseTxnRef1	BigInt	Purse Transaction Reference
PurseTxnRef2	BigInt	Purse Transaction Reference
ProductContributionValue	Money	Value of Product contribution to transaction (if Product used in transaction)
PassRemainingValue	DateTime	Expiration Date of Pass (0 if TransType <> ttPass)
MultirideRemainingValue	Int	Remaining Value (0 if TransType <> ttMultiride)
ProductTxnRef1	BigInt	Product Transaction Reference
ProductTxnRef2	BigInt	Product Transaction Reference
ProductIssuerID	Int	Product Issuer ID
ProductTypeID	Int	Product Type ID
VehicleType	Int	Vehicle Type <sup>3</sup>
PassengerType	Int	Passenger Type <sup>4</sup>
CommercialID	Int	Commercial ID read from RFCS card
UsageID	Int	FK link to Usage.UsageID.

#### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	FareDeducted	Values in this table correspond to the meaning supplied by the Regional Fare Card System software vendor for the Status field in the NOTIFY_GAK_PAYMENT_DETAIL message described in SEA-01356
1	FareNotDeducted	
2	CardNotPresent	
3	DifferentCard	
4	Timeout	
5	MultipleCardsDetected	
6	RouteIDInvalid	
7	TripIDInvalid	
8	PassengerTypeInvalid	
9	PassengerVehicleTypeInvalid	
10	ProductInvalid	
11	CardBlocked	
12	CardExpired	
13	CardInvalid	
14	Passback	

#### <sup>2</sup> Transaction Type Values

Value	Gateway Constant Name	Description
0	ttPurse	A purse product was used for redemption
1	ttPass	A pass product was used for redemption
2	ttMultiride	A multiride product was used for redemption
3	ttPurseReversal	A redemption of a purse product was voided
4	ttProductReversal	A redemption of a pass or multiride product was voided

#### <sup>3</sup> Vehicle Type Values

Value	Gateway Constant Name	Description
0	None	Values in this table correspond to the meaning supplied by the Regional Fare Card System software vendor for the seaPassengerVehicleType_t data type.
1	Bicycle	
2	Motorcycle	
3	OversizedMotorCycle	
4	Under20	
5	Overheight	
6	Carpool	
7	Vanpool	

8	Other Any	
255	VehicleNotSpecified	

**4 Passenger Type Values**

Value	Gateway Constant Name	Description
1	Adult	Values in this table correspond to the meaning supplied by the Regional Fare Card System software vendor for the PassengerType_t data type.
2	Youth	
3	Senior	
4	Disabled	
5	Low Income	

## 4.40 RFCSOperations

Contains SQL version of RFCS Operations, which contain configuration parameters for the ERG RFCS smartcard system. RFCS Operations need to be defined only if using ERG's GAK smart card reader. This table is populated by DBSync, but is not otherwise used by the system.

### Columns

Column	Type	Allow Nulls	Description
RFCSOperationID	Int	N	Primary key, always unique
OperationID	Char(20)	N	Foreign key to Items.PLU, mapping Gateway item to link with ERG
RouteID	Int	Y	Identifies "Route ID" defined in ERG data
TripID	Int	Y	Identifies "Trip ID" defined in ERG data
VehicleType	Int	Y	Identifies "Vehicle Type" defined in ERG data <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKRFCSOperationID	P	RFCSOperationID	Primary Key.

<sup>1</sup> VehicleType Values

Value	Gateway Constant Name	Description
0	None	Passengers Only
1	Bicycle	Bicycle
2	Motorcycle	Motorcycle
3	OversizedMotorcycle	Oversized Motorcycle
4	Under20	Vehicle under 20'
5	OverHeight	Vehicle under 20' Overheight
6	Carpool	Carpool
7	Vanpool	Vanpool
8	Other	Other
254	Any	Any
255	VehicleNotSpecified	Not Specified

## 4.41 TicketBanks

The **TicketBanks** table contains multipark bank data for tickets. It is populated by TCON32 for each ticket, based on the information in the **BankHeaders** and **BankDetails** tables corresponding to the ticket's access code.

### Columns

Column	Type	Allow Nulls	Description
TicketBankID	Int	N	Primary key, always unique.
VisualID	Char(40)	N	Foreign key to Tickets.VisualID.
BankNo	Int	N	The detail number within the header (the first detail is bank number 1 and so on).
BankHeaderID	Int	N	Foreign key to BankHeaders.BankHeaderID.
Balance	Int	N	Balance remaining on this bank.
FirstUse	Date	Y	The Date the Bank was first used. Updated only when NULL or the default date value (1899-12-31) upon decrement of Bank Balance.
LastUse	DateTime	Y	Last date/time this bank was updated.
LastACP	Int	Y	Foreign key to ACPs.AcplD, specifying the ACP that most recently updated this bank.
UseCount	Int	Y	Number of times this bank was used.
TicketDate	DateTime	Y	Ticket date, if a date specific ticket.
Points	Integer	Y	The point balance of a given entity

### Indexes

Name	Kind	Columns	Purpose
PKTicketBanksTicketBankID	P	TicketBankID	Primary Key.
AKTicketBanksVisualIDBankNo	A	VisualID, BankNo	Necessary for proper operation of multi-park. Also, necessary to support the re-insertion of tickets once TCON32 loses and regains its database connection.
IXTicketBanksVisualID		VisualID	Used for finding all ticket banks belonging to a ticket or pass.

## 4.42 Tickets

The **Tickets** table contains all tickets referenced by software included in the Access Control System, but may be used for tickets in other implementations to increase the number of potential reports and capabilities in the system. Tickets should be added to the table at the time of sale (a system option), but may be added at other times by Access Control stations or import programs.

### Columns

Column	Type	Allow Nulls	Description
TicketID	Int	N	Primary key, always unique
VisualID	Char(40)	N	Scan code or barcode as read from scanner; combination of AccessCode and ID.
AccessCode	Int	N	Foreign key to AccessCodes.AccessCode, corresponds an access code definition specifying ticket type information with respect to <i>Access Control</i> in general (as opposed to ticket type information found in Products). The AccessCode of a ticket also affects how the ticket is used in most Access Control reports.
ID	Char(20)	N	A unique string identifying the ticket within the set of all tickets with the same AccessCode. The ID consists of the NodeNo and Serial values (below) zero padded and concatenated. The lengths are user definable, but the defaults are 3-digit node and 6-digit serial.
NodeNo	Int	N	Node number of the computer which issued the ticket
TransNo	Int	N	The sequential ID number of the transaction from which the ticket belongs. The NodeNo and TransNo may be used together to look up transaction data in the TransHdr table.
LineNum	Int	N	Always 0. This column is not currently used by the system.
Serial	Int	N	The ticket's serial number for the node and company which issued the ticket.
Company	Int	N	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
TktCode	Char(6)	N	The category and sub-category from the Chart Of Accounts.
Qty	Int	N	The number of guests to be admitted with this ticket. Regular tickets always have a quantity of 1. Group tickets will have some number of guests which was selected during the sale. Negative quantities are used in other parts of the system for returned and voided tickets, but are not used in the Tickets table.
TktIndex	Int	N	The position of this ticket within the transaction for a specific ticket type. Each ticket in a transaction is uniquely identified by combining ProductNo, FkeyNo, and TktIndex. Each ticket record of a disbursed ticket will have the same TktIndex value. These may be uniquely identified by including the serial number, if used.
Coupons	Int	N	Number of coupons printed for this ticket.
ProductNo	Int	N	Product from which the ticket was issued.
FkeyNo	Int	N	FkeyNo is combination of the ticket type's level and fkey numbers within the product: FkeyNo = (Level * 100) + Fkey.
DiscNo	Int	N	Foreign key to Discounts.DiscountID, specifying the discount applied to this ticket, or 0 for no discount.
DiscAmt	Money	N	The amount of money, in <i>base currency</i> , discounted from the ticket. If DiscNo is 0 and this column is used, this indicates that the ticket price was edited at the time of sale. In this case, DiscAmt contains the difference between the base price and the price used.
DiscIndex	Int	N	A sequence number for the discount used in the transaction. For example, if three discounts are used in a transaction, one of the discounted tickets will have a DiscIndex value of 1, another of 2, and another of 3.
DateSold	DateTime	Y	Date and time at which the ticket was issued.
Status	Int	N	A number indicating the state of the ticket as defined by the <i>Ticket Status Values</i> table below <sup>1</sup> . A ticket may also be expired, but because that state may change without intervention by the system, it must be determined, as needed, by the system.
RemainingValue	Float	N	A floating point numeric value representing the value of the ticket. Typically, this is a number of admissions or a monetary value, but may represent any unit desired. For a regular ticket, the value would be one, or some number of guests or admissions. In a debit-card implementation, the number would represent the amount of money remaining on the ticket or card.
LastUse	DateTime	Y	The date and time that the ticket was last used by a guest. More specifically, the last time the record in the Tickets table was updated after being scanned at a scanning station.
LastAcp	Int	Y	Foreign key to ACPS.AcPdID, specifying the Access Control Point number of the scanning station at which the ticket was last scanned.
UseCount	Int	Y	The number of times the ticket has been used for admission or redemption of value.
Expiration	DateTime	Y	If not null, an expiration date and time which overrides any expiration information specified in the ticket's Access record (referenced by AccessCode).
CustNo	Char(10)	Y	A string containing the account number of the customer which purchased this ticket. This column is blank if the ticket was not purchased by a customer with an account in the system, or if the ticket was added <i>dynamically</i> at the time of its first scan.
OrderNo	Int	Y	Foreign key to Orders.OrderNo, specifying the order that the ticket was placed under (or 0 for no order).
EventNo	Int	Y	Foreign key to RMEvents.EventID, specifying the event that the ticket was sold from (or 0 for no event).
Price	Money	N	Price paid for ticket, excluding taxes and additional payments made to increase the RemainingValue of the ticket.
Tax	Money	N	Total amount of tax paid with ticket. This includes tax charged upon additional payments made to increase the RemainingValue of the ticket.
Taxes	Char(8)	N	An 8 character string containing an array of Y/N flags indicating which of the 8 possible taxes are included in the Tax value.
Commission	Float	N	The commission earned for this ticket. The commission is based on the CommissionRate in the ticket's corresponding Chart of Accounts entry.
Preprinted	Bit	N	This is selected for pre-printed ticket types, indicating that the serial number (in both the ID and Serial columns) is based on a user-defined value, not the company's current sequential serial number.
UseQty	Int	N	The number of times the ticket has been used.
Act	Int	N	The Chart of Accounts number. Defined as (Company * 100000) + (Category * 100) + Sub-Category
Duplicate	Int	N	Always contains 0 (not currently used)
TicketDate	DateTime	Y	Used for Date Specific Tickets
PLU	Char(20)	Y	Foreign key to Items.PLU, this is the PLU for the ticket. This can either be in the form TICKETPPPLLFF for tickets associated with a product, or user-defined for tickets not associated with a product.
TaxMethods	Char(8)	Y	The tax methods is an 8 character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket. <sup>2</sup>
UpdateCode	Int	Y	Specifies whether or not sales data for this ticket was received from the POS <sup>3</sup> .
CustomerID	Int	Y	Populated with the internal Customer ID value when a ticket is sold in a customer-based transaction from the POS or OE. Foreign key reference to Customers.CustomerID
ResourceID	Int	Y	Foreign key to RMResources.ResourceID. If this is an event ticket, this column points to the resource that the event ticket was sold for.
CapacityID	Integer	Yes	The unique ID of the related RMCapacity record
FirstName	Varchar(30)	Y	First name of the guest associated with this ticket
LastName	Varchar(30)	Y	Last name of the guest associated with this ticket
GalaxySiteID	Int	Y	Foreign Key reference to Sites.GalaxySiteID
ContactID	Int	Y	Foreign key to CustContacts.CustContactID

GuestPhotoRequiredPictureID	Bit	Y	Flag to indicate if a photo is required.
UpgradeValue	Float	Y	Value to be used when upgrading this ticket.
PriceToken	Varchar(60)	Y	An identifier used to reference the associated external price data for future pricing requests.
Reprinted	Bit	Y	Indicates that the ticket has already been reprinted.
ReprintOnNextScan	Bit	Y	Indicates that the ticket should be reprinted on next scan at ACS.
ActivateByDate	DateTime	Y	The date by which an inactive ticket must be activated.
EndOfLifeDate	DateTime	Y	The date Galaxy estimates that the ticket will expire. This estimate is done at the time of sale.
EndOfLifeDateStatus	Int	Y	The status of the end of life date. Indicates if the end of life date has been overridden. <sup>4</sup>
EndOfLifeLockWindow	Int	Y	Number of days past the EndOfLifeDate in which the ticket will lock. -1 indicates that the ticket will not lock.

**Indexes**

Name	Kind	Columns	Purpose
PKTicketsTicketID	P	TicketID	Primary key.
IXTicketsCustNoOrderNo		CustNo, OrderNo	Previously used to speed up some order reports, may no longer be necessary.
IXTicketsOrderNo	F	OrderNo	For collecting data for the OrderEntry's Order Breakage Report.
IXTicketsAccessCodeID		AccessCode, ID	For finding and updating a ticket record using the AccessCode and ID.
IXTicketsID		ID	For updating a ticket using the ID.
IXTicketsLastUse		LastUse	For collecting the ticket usage by order and used ticket reports.
IXTicketsLastUpdate		LastUpdate	For finding tickets most recently modified.
IXTicketsSerialNodeNo		Serial, NodeNo	To retrieve a range of tickets using the Serial and NodeNo. This Index is used by the Galaxy's Ticket Cancel function to retrieve a range of tickets using VisualID
IXTicketsDateSoldStatus		DateSold, Status, RemainingValue, AccessCode, Serial	To speed up the Unused Ticket Detail ACS report. THIS IS AN OPTIONAL INDEX, AND SHOULD ONLY BE CREATED IF THE USER IS RUNNING SQL-BASED ACS REPORTS.
IXTicketsOrderBreakageReport		OrderNo, AccessCode, Status	Order Breakage Report
IXTicketsVisualIDPLULastACP		VisualID, PLU, LastAcp	Ticket Cancel Report
IXTicketsContactID	F	ContactID	Foreign key to CustContacts.CustContactID

**1 Status Values**

Value	Gateway Constant Name	Description
0	TKT_VALID	The ticket is valid
1	TKT_VOIDED	The ticket has been voided
2	TKT_RETURNED	The ticket has been returned
3	TKT_INACTIVE	Ticket is inactive. An inactive ticket cannot be used until it is activated
5	TKT_UPGRADED	Ticket was upgraded using the Ticket Upgrade function in Galaxy
6	TKT_REPLACED	Ticket was replaced using the Reprint function in Galaxy
7	TKT_REPRINTED	Ticket was reprinted using the Reprint function in Galaxy, and the "Create New VisualId When Reprinting Tickets" option in Order Entry config is enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain.
8	TKT_RENEWAL_CREDIT	Ticket was applied to a pass renewal as a renewal credit
14	TKT_NOT_PRINTED	Ticket exists in Tickets table but has not been printed. This is used for SIAE in Order Entry since ticket information must be generated at the time of payment.

**2 TaxMethod values, per character**

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis

**3 UpdateCode Values**

Value	Gateway Constant Name	Description
0	UPDATECODE_NORMAL	Normal case, ticket was added before it was used.
1	UPDATECODE_UNSOLED	The ticket was not found when first attempting to use it, and was overridden to admit the guest anyway. Doing this resulted in TCON32 adding this ticket dynamically. However, the sales data for this ticket has not yet been received. Therefore, many of the column values for this ticket are blank or zero or NULL (PLU, Price, DateSold, etc.).
2	UPDATECODE_UPDATED	The ticket was created dynamically with an UpdateCode value of 1, but eventually the late sales data was received. All column values were updated with the values from the sales data, except for those used to track the usage of the ticket (UseCount, RemainingValue, etc.).
3	UPDATECODE_OLD_USE	Similar to value 2, the ticket was created dynamically and updated with late sales data. However, the DateSold time is after the first use of the ticket, indicating a possible mismatch between the clocks of TCON32 and the POS, or a counterfeit ticket may have been used.

**4 EndOfLifeStatus Values**

Value	Gateway Constant Name	Description
0	EOL_STATUSCODE_DEFAULT	Default status for the end of life date
1	EOL_STATUSCODE_OVERRIDDEN	The end of life date has been manually overridden

## 4.43 SuperTickets

This table shows the link of all ticket records and pass records for a given customer's ticket. Throughout the life of a given ticket/pass, it may be re-issued, upgraded, downgraded etc. This table will show all links to each instance of the given ticket i.e. Ticket lineage.

### Columns

Column	Type	Allow Nulls	Description
SuperTicketID	Int	N	Primary key, always unique
NextID	Int	N	SuperTicketID for the next linked ticket in the table
VisualID	Int	N	VisualID for the this ticket
BaseID	Int	N	References the ID of the first ticket in the lineage
AuxID	Int	N	References the ID in the Pass or Tickets table
Status <sup>1</sup>	Int	N	Current status of this ticket
TicketType <sup>2</sup>	Int	N	Type of ticket
CurrentRecord	Bit	N	Flag to show if this is the current ticket in the lineage. This field is not used at all for packages or package details. It is generally not used for decision making in Galaxy.
ContactID	Integer	Y	Foreign Key, references the unique identifier (CustContactID) of the CustContacts SQL table
ReplenishStatus <sup>3</sup>	Integer	Y	Status of auto-replenish
IssuanceStatus	Integer	Yes	Status of if the entitlement has been issued. Currently only 0 for issued and 1 for not issued. This could be more later.
PackageDetailID	Integer	Yes	References PackageDetailID on PackageDetails table.
Deferred	Bit	Yes	Indicates if the SuperTicket record represents a package detail that was sold as a deferred entitlement.

### Indexes

Name	Kind	Columns	Purpose
PKSuperTicketsSuperTicketID	PK	SuperTicketID	Primary Key.
IXSuperTicketsVisualIDTktType	IX	VisualID, TicketType	Index to look up by VisualID or VisualID and TicketType
IXSuperTicketsBaseID	IX	BaseID	Needed to query lineage faster
IXSuperTicketsContactID	IX	ContactID	Speedup Queries on ContactID
IXSuperTicketsAuxID	IX	AuxID, TicketType	Search by TicketType and AuxID
IXSuperTicketsVisualID	IX	VisualID	To improve performance of selects by VisualID

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	ST_ACTIVE_STATUS	Active - Valid Record
1	ST_INACTIVE_STATUS	Inactive - A record that was produced as inactive and has not been activated yet
2	ST_DEACTIVE_STATUS	Deactivated - a record that was active and has been deactivated. Can never be activated again.
3	ST_REPLACED_STATUS	Replaced - the physical media was replaced with a new Visual ID
4	ST_UPGRADED_STATUS	Upgraded - the record was upgraded to a new one. A new Visual ID was assigned
5	ST_RETURNED_STATUS	Ticket has been returned
6	ST_VOID_STATUS	Ticket has been voided
7	ST_REPRINTED_STATUS	Reprinted - The SuperTicket was reprinted using the Reprint function in Galaxy, and the "Create New VisualId When Reprinting Tickets" option in Order Entry config is enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain.

### <sup>2</sup> TicketType Values

Value	Gateway Constant Name	Description
0	ST_TICKET_TYPE	AuxID links to a row in the Tickets table
1	ST_PASS_TYPE	AuxID links to a row in the Pass table
2	ST_PACKAGE_TYPE	This is a package header record. AuxID is a JnlPackageID from the JnlPackages table.
3	ST_ITEM_TYPE	This is an item record. AuxID is a JnlDetailID from the JnlDetails table.
4	ST_GTSDEBIT_TYPE	This is a GTS debit card. AuxID is a DebitCardID from the DebitCards table.
5	ST_DEBIT_TYPE	(not currently used). This is a debit card that is not a GTS debit type.

### <sup>3</sup> ReplenishStatus Values

Value	Gateway Constant Name	Description
0	ST_NOT_REPLENISH_STATUS	Replenish is not applicable
1	ST_ACTIVE_REPLENISH_STATUS	Replenish is active
2	ST_USED_REPLENISH_STATUS	Replenish is used
3	ST_CANCEL_REPLENISH_STATUS	Replenish is cancelled
4	ST_SUSPEND_REPLENISH_STATUS	Replenish is suspended

## 4.44 Usage

The **Usage** table records Access Control activity. This includes the scanning of all tickets, passes, and other barcoded media at Access Control Points, the logging on and off of ticket takers at Access Control Points, the void and return of tickets used in the Access Control System, and the entries and exits through turnstiles and other counting devices. The **Usage** table is used for most, if not all Access Control reports and is used for tracking the history of individual tickets and passes. References to "the ticket" in the following section imply the barcoded ticket which was scanned to produce the Usage record. Note that scanning a ticket is not the only way to produce a Usage record, and those fields which reference "the ticket" are not used for non-ticket Usage (such as logon/logoff records and turnstile counts).

### Columns

Column	Type	Allow Nulls	Description
UsageID	Int	Y	Value is either from GatewayCounters or a SQL sequence, depending on the database configuration. It is sequentially incremented for each row. <b>NOTE:</b> There is no Unique key/index on this column. <b>However</b> , if the @CreatePrimaryKeyOnUsageTable flag is turned on when the main Galaxy database script is run, this column will become a NOT NULL, primary key field and any NULL values will be populated with values from the Gateway Counter or SQL sequence.
UseTime	DateTime	N	The time at which activity occurred at the ACP.
VisualID	Varchar(40)	N	The visual ID from the scanned ticket or pass.
AccessCode	Int	N	The AccessCode extracted from the barcode scan which generated the Usage record.
IDNo	Char(40)	N	The barcode as read from the scanner, same as VisualID.
Code <sup>2</sup>	Int	N	The type of activity Valid values for this column are shown in the <i>Code Values</i> table.
ACP	Int	N	The ID number of the Access Control Point at which the activity took place.
Status <sup>1</sup>	Int	N	The result of the activity, as defined in the <i>Status Values</i> table.
Qty	Int	N	The number of guests admitted or value redeemed.
UseNo	Int	N	The number of time the ticket or pass was used.
Operator	Int	N	Foreign key into GXUers.UserID, referencing the user logged on at the time the Usage record was created. This value is 0 if nobody was logged-on.
EntryMethod <sup>3</sup>	Int	N	A number representing the method used to enter the barcode into the system which in-turn generated the Usage record. Possible values of EntryMethod are defined in the <i>EntryMethod Values</i> table below
SerialNo	Char(20)	N	The ID portion of the ticket which was scanned to produce the Usage record.
Override	Int	N	If the operator overrides an invalid ticket and lets the guest into the park, this field stores the original reason the ticket was rejected <sup>1</sup> .
UsageCondition <sup>4</sup>	Int	Y	A number representing the condition of TCon32 when the ticket was scanned. Possible values of UsageCondition are defined in the <i>UsageCondition Values</i> table below.
OriginalStatus	Int	Y	A number representing the value of the Usage Status, before running the Offline Recategorization program, "ResetUse".
BankNo	Int	Y	The bank detail number validated by ACS2. In other words, the bank detail that was valid and allowed the admission. This bank number is sent to TCON32 which uses it to determine the correct ticket bank detail to remove uses from.
BiometricStatus <sup>5</sup>	Int	Y	Status of Biometric activity performed during this usage
OperationID	Int	Y	Operation that was used
Points	Int	Y	The number of points debited from a stored value points card
GalaxySiteID	Int	Y	Foreign Key reference to Sites.GalaxySiteID
ScannedVisualID	Varchar(40)	Y	The visual ID of the barcode that is scanned. This may differ from the visual ID if the scanned barcode is from a SuperTicket record.
AttractionID	Int	Y	The Attraction that was configured on the ACP where this usage occurred. Foreign key to Attractions.IDNo.
OriginalUsageID	Int	Y	The original usage id that this usage was based off of. Used for Reversals and Voids of usages.
FacilityID	Int	Y	Foreign key to Facility.IDNo. ACPS can now function as an exit of a facility. If zero, the facility that the usage is applied to can be obtained via Usage.ACPS. If non-zero, the usage applies to the indicated facility.
EntitlementCharged	Bit	Y	If 1, the entitlement has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION.
BankCharged	Bit	Y	If 1, the bank has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION, and the same bank was used for both admissions.
NodeNo	Int	Y	The node number this activity occurred at.
TransNo	Int	Y	The POS transaction number this activity occurred in (if available).
PLU	nVarChar	Y	The PLU of the scanned item (if available).
ExternalUsage	bit	Y	Indicates if the usage originated from an external validation.
ACSRReservationConfirmationNumber	nvarchar(50)	Y	The confirmation number of the reservation used during validation.
ID	BigInt	N	Unique Primary Key field
ProcessingError <sup>6</sup>	Int	Y	<p>If not null and not zero, indicates that a validation or processing error occurred while this usage was being examined by the Usage Processor component in Direct-to-Database ACS32/Admission Control at POS or TCON32.</p> <p>Many of these errors can only occur if a scan occurs while the database is offline, since the errors would normally result in a ticket not being valid for admission.</p> <p>The purpose of this column is to report usage processing errors that are related to offline scanning, and to configuration errors, not invalid scans or operator overrides:</p> <ul style="list-style-type: none"> <li>- Most tests are not performed for invalid scans. The Usage.Status column can be used to report invalid scans.</li> <li>- Most tests are not performed when invalid validation results are overridden by operators. Overrides can be found by examining Usage.Override. The conditions under which each test is performed are noted in the associated values table below.</li> </ul> <p>The Exceptions Live Access report displays usages with non-null processing errors. These can be compared to overridden scans in the same report.</p> <p><b>NOTE:</b> Only one ProcessingError value is stored in the Usage table. Checks are performed in the order below. If any check fails, additional checks are not performed.</p> <ol style="list-style-type: none"> <li>1. PE_INVALID_ACP</li> <li>2. PE_INVALID_MEDIA_DEF</li> <li>3. PE_RFIDMAP_NOT_FOUND</li> <li>4. PE_INVALID_ACCESS_CODE</li> </ol>

			5. PE_NO_UPDATE_TABLE 6. PE_TICKET_NOT_FOUND, PE_PASS_NOT_FOUND, PE_DEBIT_CARD_NOT_FOUND 7. PE_INVALID_SUPER_TICKET_NEXT_ID 8. Status errors
UsageGUID	UniquedIdentifier	Y	GUID associated to this usage. Use for joining a Usage row to a JnlUsage row.

**Indexes**

Name	Kind	Columns	Purpose
IXUsageAccessCodeIDNo		AccessCode, IDNo	Unknown, might not still be necessary.
IXUsageTimeACodeStatusQtyACP		UseTime, AccessCode, Code, Status, Qty, ACP	This index is used to improve the performance of loading usage counts.
IXUsageVisualID		VisualID	Improves performance of finding usage by barcode, for example in the Ticket Cancel function.
IXUsageEntryMethod		EntryMethod	To speed up the Scanning ACS report. THIS IS AN OPTIONAL INDEX, AND SHOULD ONLY BE CREATED IF THE USER IS RUNNING SQL-BASED ACS REPORTS.
IXUsageSerialNo		SerialNo	This index is used for displaying pass usage (the SerialNo field corresponds to a pass's account).
IXUsageID	IX	UsageID	<b>Index Note:</b> This index can become a Primary Key if the @CreatePrimaryKeyOnUsageTable flag is set in the GalaxyDatabase.sql file.
IXUsageScannedVisualID	A	Scanned Visual ID	To allow the usage table to be searched by the scanned visual ID.
IXOriginalUsageIDUsageID	A	OriginalUsageID, UsageID	Improves performance of finding usage by UsageID, for example while searching for usages in the Usage Manager for reversal function.
IXUsageUsageGUID	IX	UsageGUID	Improve performance when finding matching JnlUsage and Usage rows.

**<sup>1</sup> Status Values**

Value	Gateway Constant Name	Description
0	FACILITY_ADMISSION	The ticket was valid and accepted for admission.
1	TICKET_EXPIRED	The ticket has no uses remaining, the pass is expired, or the date of scan was later than the date specified in the barcode.
2	TICKET_NOT_VALID	The ticket did not meet Valid Date record, Expiration column, or any other date/time restrictions.
3	TICKET_NUMBER_UNKNOWN	The ticket was not found in the Tickets table and is not dynamic.
4	TICKET_FORMAT_UNKNOWN	The format of the barcode was not recognized.
5	TICKET_ACCESS_UNDEFINED	The AccessCode specified by the barcode was not valid.
6	FACILITY_REENTRY	The ticket was found in the ReEntry table.
8	TICKET_PROCESSING	Validation is still being performed.
9	TICKET_RANGE_UNKNOWN	The ticket's Access Definition specified Range checking, but the ticket was not found in the TktRange table.
10	TICKET_AGENT	The entry is a record of a ticket taker logon or logoff.
11	TICKET_LOCKED_OUT	The ticket was found in the Lockout table.
12	TICKET_RETURNED	The ticket had been returned.
13	TICKET_VOIDED	The ticket had been voided.
14	PASS_USED_TODAY	A pass with only one use per day allowed had already been used today.
15	PASS_NOT_VALID	The pass was not found in the Pass table.
16	PARTY_CANCELLED	The party prompt was cancelled.
17	PROMPT_CANCELLED	A prompt was cancelled.
18	INVALID_FACILITY_REENTRY	A reentry was detected but not allowed.
19	CCF_APPLY_NOW	When ACS2 needs to apply a CCF.
20	DATABASE_ERROR	Could not connect to Usagelog (Usagelog.Dat).
21	PASS_WAS_REPLACED	Pass has been reissued (and is not valid).
22	REENTRY_CANCELLED	The cancel button was pressed at a reentry prompt.
23	TICKET_CROSSOVER	The entry was a valid crossover.
24	TICKET_INVALID_CROSSOVER	The entry was an invalid crossover.
25	INSUFFICIENT_FACILITYUSES	The ticket had no valid banks with a balance greater than zero.
26	INVALID_FACILITY	The ticket's operations did not match any active operations at the ACP.
27	ADMISSION_NOT_ACCEPTED	This turnstile does not accept admissions
28	REENTRY_NOT_ACCEPTED	This turnstile does not accept reentries
29	CROSSOVER_NOT_ACCEPTED	This turnstile does not accept crossovers
30	TICKET_FREE_ENTRY	Valid status pass scan for a free entry to a park. The free entry calendar and the free entry max party determine a pass scan with a party is a free entry or not. A free entry pass scan does not increase the Passes.UseCount.
31	NOUSESLEFT	No uses left on Multi-park Pass.
32	TICKET_LOCKED	Ticket was locked by another ACP when the validation occurred
33	TICKET_INACTIVE	The ticket is/was in an inactive state (i.e., has not been activated yet)
34	OVERRIDE_NOT_ALLOWED	Overrides not allowed on prefetched tickets where fetch failed
35	TICKET_UPGRADED	Ticket or pass has been upgraded
36	TICKET_REPLACED	Ticket has been replaced
37	EXECUTE_FUNCTION	This scan is to perform a function key task
38	REGISTRATION_FAILED	Biometric scan failed to register its scan for future biometric scan validation
39	IDENTIFICATION_FAILED	Biometric scan failed to find a match
40	TICKET_VELOCITY_LOCKED	A ticket is invalid because it is locked according to the rules of the velocity subsystem in ACS32 and TCON32
41	TICKET_REPRINTED	The ticket or pass has been reprinted by a node with the "Create New VisualID When Reprinting Tickets" option in Order Entry config enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain. Regular ticket validation is also aborted with this status if the Status column in Tickets or Passes is set to TKT_REPRINTED or PASS_REPRINTED, respectively.
42	GENERATED_NEW_TICKET	This usage was created only to add the original ticket to the reentry table when supplemental tickets were generated. No counts have been updated. This status will appear with usage records with a code of USAGE_REENTRY_TABLE (6).
43	TICKET_BLOCKED	

44	DEBIT_CARD_NOT_FOUND	
45	INSUFFICIENT_FACILITY_POINTS	Insufficient points on debit card
46	POINTS_COMBINED	Valid - points combined with another ticket
47	TICKET_ALREADY_COMBINED	
48	PASS_IS_UNISSUED	
49	PHOTO_REJECTED	This usage indicates that the user was displayed a photo verification prompt, and rejected the photo as part of the current scan. This only applies when the option "On rejected photo, invalidate pass" is selected on the Access Code.
50	PHOTO_REJECTED_PREVIOUSLY	This usages indicates that the pass currently being validated has a photo that was rejected as part of an earlier scan by the user. This only applies when the option "On rejected photo, invalidate pass" is selected on the access code.
51	BANK_REQUIREMENT_NOT_SATISFIED	The Bank Requirement was not satisfied.
52	PHOTO_NOT REVIEWED	This indicates that the status of the pass (and corresponding photo) is set to PASS_NOT REVIEWED when scanned at the ACP
53	TICKET_NOT_PRINTED	Indicates the Ticket or Pass exists in the database, but has not yet been printed. This status occurs when SIAE tickets or passes are paid for in Order Entry but not issued.
54	ALL_MEMBERS ADMITTED	The pass is invalid because all adults, children and guests were previously admitted in the same business day.
55	INVALID_ATTRACTION	The ticket is not valid for the attraction because one or more required operations are not activated at the ACP it was scanned at. Analogous to INVALID_FACILITY.
56	INVALID_ATTRACTION REENTRY	The ticket is not valid for reentry to the attraction. Analogous to INVALID_FACILITY REENTRY.
57	INSUFFICIENT_ATTRACTIONUSES	The ticket's use balance is insufficient for admission to the attraction. Analogous to INSUFFICIENT_FACILITYUSES.
58	INSUFFICIENT_ATTRACTION_POINTS	The ticket's point balance is insufficient for admission to the attraction. Analogous to INSUFFICIENT_FACILITY_POINTS.
59	GUEST_NOT_MOVING	This status is used internally by the Admission Control Validator to represent that the guest is not moving relative to the facility. Usage is not recorded in this case. This status will never appear in the Usage, Reentry or JnlUsage tables.
60	ATTRACTION_ADMISSION	The guest was admitted to an attraction. Analogous to FACILITY_ADMISSION. This is a valid status.
61	ATTRACTION_REENTRY	The guest re-entered an attraction. Analogous to FACILITY REENTRY. This is a valid status.
62	ATTRACTION_CROSSOVER	The guest is performing a crossover from one attraction to another. Analogous to TICKET_CROSSOVER, but for an attraction.
63	INVALID_ATTRACTION_CROSSOVER	The guest attempted to crossover from one attraction to another, but the movement was not allowed by the rules of the bank. Analogous to TICKET_INVALID_CROSSOVER, but for an attraction.
64	VEHICLE_NOT_PRESENT	A vehicle is required but not present.
65	ACP_NOT_DEFINED	The requesting ACP is not defined in ACPS.
66	GUEST_MOVEMENT_NOT_ALLOWED	Multi-park guest movement is not allowed
67	ATTRACTION_ONLY_TICKET	The ticket can only be used at attractions
68	PASS_IS_INACTIVE	The pass is/was in an inactive state (i.e., has not been activated yet).
69	GUEST_PHOTO_REJECTED	This usage indicates that the user was displayed a guest photo verification prompt, and rejected the photo as part of the current scan.
70	CONTACT_DECEASED	Indicates that the Pass's contact is deceased (i.e. CustContact.Deceased).
71	REMAINING_USES_EXCEED_DAYS	The remaining uses is greater than the number of days left to use the ticket.
72	INVALID_REQUEST	Invalid data was supplied in the request.
73	RFID_NOT_FOUND	RFIDMaps row not found.
74	UPDATE_TABLE_NOT_SPECIFIED	AccessCodes Update Table not specified.
75	PLUGIN_NOT_SUPPORTED	Prefetch plugin not supported in Lookup Proc.
76	RESERVATION_REQUIRED	A reservation is required and was not found.
77	GUEST_NAME_REJECTED	This usage indicates that the user was displayed a guest name verification prompt, and rejected the name as part of the current scan.
78	SERVICE_API_ERROR	An ACSService API error occurred during validation.
79	DOUBLE_SCAN	Generated by the ticket validator in ACS32 when a double scan is detected and the "Record status as DOUBLE_SCAN instead of INVALID_FACILITY REENTRY" is enabled in the ACS configuration editor.
80	GATE_FRAUD_ENTRY_ZONE_INTRUSION	Gate's entry sensors have been obscured too long during validation.
81	GATE_FRAUD_EXIT_ZONE_INTRUSION	Gate's exit sensors have been obscured too long during validation.
82	GATE_FRAUD_BLOCKED_AISLE	Person entered a locked side.
83	GATE_FRAUD_LONG_TRANSIT	Person took too long to end the transit.
84	GATE_FRAUD_TAILGATING	Tailgating has been detected during passage.
85	GATE_FRAUD_WRONG WAY	Wrong Way passage has been detected.
86	GATE_FRAUD_CRAWL	Detected attempt to crawl under the doors.
87	GATE_FRAUD_DOOR_FORCED	Detected attempt to force the doors open.
88	GATE_FRAUD_JUMP_OVER	Detected attempt to pass over the doors.
89	GATE_FRAUD_TRAPPING	Detected attempt to block the closing of the doors.
90	GATE_FRAUD_TURNING_BACK	Guest turned back after obscuring the last sensor.
91	GATE_FRAUD_GENERAL	Fraud status for general gate fraud that doesn't fall into another category.

## <sup>2</sup> Code Values

Value	Gateway Constant Name	Description
0	USAGE_SCAN	A barcode other than a <i>logon card</i> was scanned
1	USAGE_VOID	The usage record was voided
2	USAGE_LOGON	A ticket taker logged on
3	USAGE_LOGOFF	A ticket taker logged off
4	USAGE_COUNT	The record is a turnstile count
5	USAGE_BREAKAGE	Revenue was posted after order expired  This is AX1180 only. Galaxy does not create this Usage record.
6	USAGE_REENTRY_TABLE	This usage will be used only to add the scanned ticket to the reentry table. Use counts will not be modified.
7	USAGE_SMARTCARD	Smartcard swipe
8	USAGE_SMARTCARD_DATA	Internal data for smartcard, not stored in Usage - No longer used
9	USAGE_ADJUSTMENT	For adjusted usage

10	USAGE_SMARTCARD_VOID	Corresponds to a void of a USAGE_SMARTCARD row as USAGE_VOID corresponds to a USAGE_SCAN row
12	USAGE_MEMBER	This usage is for a member. The Contact ID of the member is stored in the MemberUsage table. An associated usage record with Code = USAGE_SCAN is created for all members admitted at the same time.
13	USAGE_ATTRACTION	This usage represents the usage of an attraction resource. This is analogous to USAGE_SCAN.
14	USAGE_ATTRACTION_VOID	This usage represents a void of a usage of an attraction resource. This is analogous to USAGE_VOID.
15	USAGE_ATTRACTION_REVERSAL	This usage represents a reversal of a usage of an attraction resource. This is analogous to USAGE_REVERSAL.
16	USAGE_VEHICLE	This record represents a vehicle usage.
17	USAGE_GATE_TIMEOUT	This record represents a gate timeout which will trigger a usage reversal.
18	USAGE_GATE_FRAUD	This record represents fraud detected by the gate. Type of fraud can be determined from the status.

**3 EntryMethod Values**

Value	Gateway Constant Name	Description
0	ENTRY_MANUAL	A ticket barcode was entered or a ticket taker logged on or off by typing data into a terminal or PC keyboard.
1	ENTRY_SCANNED	A ticket or <i>logon card</i> barcode was scanned into the system.
2	ENTRY_FKEY	A predefined barcode was entered into the system using a function key.
3	ENTRY_MAGNETIC	The barcode was scanned from a magnetic strip on the ticket.
4	ENTRY_SMARTCARD	The barcode was received from a smart card reader.
5	ENTRY_RFID	The barcode was read from a radio frequency receiver.
6	ENTRY_JR_UNIT	The barcode was read from a JR ticket reader.
7	ENTRY_IMPORTED	The barcode was imported.
8	ENTRY_TRANSACTION_MANUAL	Manually entered as part of POS transaction
9	ENTRY_TRANSACTION_SCAN	Scanned as part of POS transaction
255	ENTRY_NONE	The method used to enter the barcode is not known.

**4 UsageCondition Values**

Value	Gateway Constant Name	Description
0	ucONLINE	The ACP and TCon32 were online when the ticket was scanned.
1	ucOFFLINE	Either the ACP or Tcon32 was offline when the ticket was scanned.
2	ucOFFLINE_VERIFIED	(not used)
3	ucOFFLINE_RECATEGORIZED	When TCON32 receives an offline usage (usage with UsageCondition ucOFFLINE) and the "Re-evaluate offline scans" option is enabled in the TCON32 configuration, the UsageCondition column is set to this value if any column in that usage is modified by TCON32 during the re-evaluation process.
4	ucSYSTEM_ADJUSTMENT	When usage is modified by the System Adjustment Galaxy function, the UsageCondition column is set to this value.

**4.44.1<sup>5</sup> BiometricStatus Values**

Value	Gateway Constant Name	Description
0	bsNONE	No meaning. Biometric activities were not performed or do not apply to this validation.
1	bsDISABLED	This ticket should have had Biometric activities performed against it, but the Biometric system was disabled by the operator using the TOGGLE BIOMETRIC Admission Control Function.
2	bsREGISTERED	Biometric information was successfully registered for this ticket.
3	bsIDENTIFIED AGAINST TICKET	The finger image was successfully identified against the ticket in the Biometric database.
4	bsIDENTIFIED AGAINST TRANSACTION	The finger image was successfully identified against another ticket in the same transaction in the Biometric database.
5	bsREGISTER_CANCELLED	A biometric registration should have completed but was cancelled by the operator.
6	bsREGISTER_FAILED	The biometric registration failed. This ticket was rejected by the Validator. UsageLog.Status in this case will be REGISTRATION FAILED.
7	bsREGISTER_ERROR	An error occurred during registration. The system didn't know how to handle the error, so this status was recorded, and the guest allowed admission.
8	bsIDENTIFY_CANCELLED	A biometric identification should have completed but was cancelled by the operator.
9	bsIDENTIFY_FAILED	The biometric identification failed. This ticket was rejected by the Validator. UsageLog.Status in this case will be IDENTIFICATION FAILED.
10	bsIDENTIFY_ERROR	An error occurred during identification. The system didn't know how to handle the error, so this status was recorded, and the guest allowed admission.

**4.44.2<sup>6</sup> ProcessingError Values**

Value	Gateway Constant Name	Description
0	PE_NO_ERROR	NULL or 0 indicates that no error occurred. Errors 100-299 are reserved for invalid values for configuration databases, or incorrect data in the database
100	PE_INVALID_MEDIA_DEF	Indicates that the Usage Processor could not find a Media Definition matching the scan in Usage.Scan.  <i>Conditions under which the test is performed:</i>  This test is performed only for usages that are valid for guest admission.  <i>Possible causes of this status:</i>  TCON is being used to process usage, and a media definition is not defined in the SQL Media table, but is defined in an ACP's local Media BTree file.  An RFID serial number was scanned, and the RFID serial number matches an RFID media definition, but when the Visual ID for the RFID serial number was loaded from the RFIDMaps table, a media definition could not be located for that Visual ID.
101	PE_INVALID_ACCESS_CODE	An Access Code matching the Media Definition and Visual ID could not be found in the AccessCodes table.  <i>Conditions under which the test is performed:</i>  This test is performed only for usages that are valid for guest admission, where the validation was not overridden by an operator.  <i>Possible causes of this status:</i>

		TCON is being used to process usage, and an access code is not defined in the SQL AccessCodes table, but is defined in an ACP's local Access BTree file.
102	PE_INVALID_ACP	<p>The ACP that generated the usage was not found in the ACPs table.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for all usages generated by any ACP, including LOGON, LOGOFF, and any other usage record.</p> <p><i>Possible causes of this status:</i></p> <p>TCON is being used to process usage, and TCON's ACPs table (obtained from SQL), does not match an ACP's local ACPs table at an ACP.</p>
103	PE_INVALID_SUPER_TICKET_NEXT_ID	<p>This error is generated if, while processing SuperTicket usage, the Usage Processor cannot load a SuperTicket because a SuperTicket.NextID column value points to a SuperTicket row that does not exist in the database.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission, and for Void, Reversal and Adjustment usage.</p> <p>This test is performed as each SuperTicket is examined by the Usage Processor. SuperTickets are examined in order as the Usage Processor is searching for a SuperTicket to deduct a balance from. Once the Usage Processor finds a SuperTicket to deduct it stops searching the chain.</p> <p><i>Possible causes of this status:</i></p> <p>This error indicates that the structure of the SuperTickets table is not valid. Normally this should never occur.</p>
		Errors 300-599 are reserved for data configuration errors
300	PE_NO_UPDATE_TABLE	<p>A Media Def and Access Code was found, but the Media Def Lookup By setting is set to (Default), and an update table is not specified in the Access Code's Validate, Pass or StoredValue columns, so the database cannot be updated for this usage.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed only for usages that are valid for guest admission.</p> <p><i>Possible causes of this status:</i></p> <p>Either the Media Definition or the Access Code is configured improperly, or the Usage does not correspond to anything in any table, in which case this should not be considered an error.</p>
		Errors 600-999 are reserved for existence errors
600	PE_TICKET_NOT_FOUND	<p>The Usage matches a Media Def and Access Code. The Access Code specifies that it is a Ticket and should exist in the Tickets table, but a Ticket with a matching Visual ID was not found in that table.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission, as long as the validation result was not overridden by an operator.</p> <p>This test is not performed if the "Dynamic" option in the Access Code is enabled.</p> <p>This test is also performed for Void, Reversal and Adjustment Usage, but should not fail for such usage unless the backing entitlement was deleted from the database after the void, reversal or adjustment occurred.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p> <p>This error can also occur if the POS sales information for a ticket has not been sent to the database by the POS's Ticket Sender service before the ticket is scanned at a turnstile. The ticket still needs to be scanned in an offline state for this ProcessingError to be reported.</p> <p>If the Ticket Sender updates the POS sales information for a ticket it also checks for any Usage of that ticket with ProcessingError = PE_TICKET_NOT_FOUND. If any usage is found, the Ticket Sender resets the ProcessingError column to NULL.</p>
601	PE_PASS_NOT_FOUND	<p>The Usage matches a Media Def and Access Code specifying that it is a Pass and should exist in the Passes table, but a Pass with a matching Visual ID was not found in that table.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission, as long as the validation result was not overridden by an operator.</p> <p>This test is also performed for Void, Reversal and Adjustment Usage, but should not fail for such usage unless the backing entitlement was deleted from the database after the void, reversal or adjustment occurred.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
602	PE_DEBIT_CARD_NOT_FOUND	<p>The Usage matches a Media Def and Access Code specifying that it is a Debit Card and should exist in the DebitCards table, but a Debit Card with a matching Visual ID was not found in that table.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission, as long as the validation result was not overridden by an operator.</p> <p>This test is also performed for Void, Reversal and Adjustment Usage, but should not fail for such usage unless the backing entitlement was deleted from the database after the void, reversal or adjustment occurred.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
603	PE_RFIDMAP_NOT_FOUND	<p>A Media Def was found, and the Media Def specifies that the Visual ID refers to a row in the RFIDMaps table, but a row with that RFID number was not found in that table.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that valid for guest admission and occurred while the system was offline.</p>

		<p><i>Possible causes of this status:</i></p> <p>This error occurs if an RFID Card is scanned in an offline state and the RFID Card is not present in the database, or the RFIDMaps row has a blank Visual ID.</p>
		Errors 1000-1999 are reserved for ticket validity errors
1000	PE_TICKET_VOIDED	<p>Scan usage arrived for a ticket that has been voided, i.e. the Tickets.Status column contains the value TKT_VOIDED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a ticket is voided before previously-generated usage for that ticket is processed by the Usage Processor.</p>
1001	PE_TICKET_RETURNED	<p>Scan usage arrived for a ticket that has been returned, i.e. the Tickets.Status column contains the value TKT_RETURNED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a ticket is returned before previously-generated usage for that ticket is processed by the Usage Processor.</p>
1002	PE_TICKET_INACTIVE	<p>Scan usage arrived for a ticket that has not been activated, i.e. the Tickets.Status column contains the value TKT_INACTIVE.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
1003	PE_TICKET_UPGRADED	<p>Scan usage arrived for a ticket that has been upgraded, i.e. the Tickets.Status column contains the value TKT_UPGRADED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a ticket is upgraded before previously-generated usage for that ticket is processed by the Usage Processor.</p>
1004	PE_TICKET_REPLACED	<p>Scan usage arrived for a ticket that has been replaced, i.e. the Tickets.Status column contains the value TKT_REPLACE.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a ticket is replaced before previously-generated usage for that ticket is processed by the Usage Processor.</p>
1005	PE_TICKET_REPRINTED	<p>Scan usage arrived for a ticket that has been reprinted with a new Visual ID, i.e. the Tickets.Status column contains the value TKT_REPRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a ticket is reprinted before previously-generated usage for that ticket is processed by the Usage Processor.</p>
1006	PE_TICKET_NOT_PRINTED	<p>Scan usage arrived for a ticket that exists in the Tickets table but was never printed, i.e. the Tickets.Status column contains the value TKT_NOT_PRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
		Errors 2000-2999 are reserved for pass validity errors
2000	PE_PASS_VOIDED	<p>Scan usage arrived for a pass that exists in the Passes table but has been voided, i.e. the Passes.Status column contains the value PASS_VOIDED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is voided before previously-generated usage for that pass is processed by the Usage Processor.</p>
2001	PE_PASS_RETURNED	<p>Scan usage arrived for a pass that exists in the Passes table but has been returned, i.e. the Passes.Status column contains the value PASS_RETURNED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p>

		<p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is returned before previously-generated usage for that pass is processed by the Usage Processor.</p>
2002	PE_PASS_REPLACED	<p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is replaced before previously-generated usage for that pass is processed by the Usage Processor.</p>
2003	PE_PASS_UPGRADED	<p>Scan usage arrived for a pass that exists in the Passes table but has been upgraded, i.e. the Passes.Status column contains the value PASS_UPGRADED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is upgraded before previously-generated usage for that pass is processed by the Usage Processor.</p>
2004	PE_PASS_REPRINTED	<p>Scan usage arrived for a pass that exists in the Passes table but has been reprinted with a new Visual ID, i.e. the Passes.Status column contains the value PASS_REPRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is reprinted before previously-generated usage for that pass is processed by the Usage Processor.</p>
2005	PE_PASS_BLOCKED	<p>Scan usage arrived for a pass that exists in the Passes table but is marked blocked, i.e. the Passes.Status column contains the value PASS_BLOCKED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is blocked before previously-generated usage for that pass is processed by the Usage Processor.</p>
2006	PE_PASS_UNISSUED	<p>Scan usage arrived for a pass that exists in the Passes table but has not been issued, i.e. the Passes.Status column contains the value PASS_UNISSUED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
2007	PE_PASS_REJECTED	<p>Scan usage arrived for a pass that exists in the Passes table but its photo status is set to "rejected", i.e. the Passes.Status column contains the value PASS_REJECTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is marked rejected before previously-generated usage for that pass is processed by the Usage Processor.</p>
2008	PE_PASS_NOT REVIEWED	<p>Scan usage arrived for a pass that exists in the Passes table but its photo status is set to "not reviewed", i.e. the Passes.Status column contains the value PASS_NOT REVIEWED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a pass is marked not reviewed before previously-generated usage for that pass is processed by the Usage Processor.</p>
2009	PE_PASS_NOT_PRINTED	<p>Scan usage arrived for a pass that exists in the Passes table but was never printed, i.e. the Passes.Status column contains the value PASS_NOT_PRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
		Errors 3000-3999 are reserved for DebitCard errors
3000	PE_DEBIT_CARD_VOIDED	Scan usage arrived for a debit card that exists in the DebitCards table but has been voided, i.e. the DebitCards.Status column contains the value DEBIT_CARD_VOIDED.

		<p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a debit card is voided before previously-generated usage for that debit card is processed by the Usage Processor.</p>
3001	PE_DEBIT_CARD_RETURNED	<p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a debit card is returned before previously-generated usage for that debit card is processed by the Usage Processor.</p>
3002	PE_DEBIT_CARD_EXPIRED	<p>Scan usage arrived for a debit card that exists in the DebitCards table but has been expired, i.e. the DebitCards.Status column contains the value DEBIT_CARD_EXPIRED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a debit card expires before previously-generated usage for that debit card is processed by the Usage Processor.</p>
3003	PE_DEBIT_CARD_INACTIVE	<p>Scan usage arrived for a debit card that exists in the DebitCards table but has been inactive, i.e. the DebitCards.Status column contains the value DEBIT_CARD_INACTIVE.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>
3004	PE_DEBIT_CARD_REPRINTED	<p>Scan usage arrived for a debit card that exists in the DebitCards table but has been reprinted, i.e. the DebitCards.Status column contains the value DEBIT_CARD_REPRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state. This can also occur if a debit card is reprinted before previously-generated usage for that debit card is processed by the Usage Processor.</p>
3005	PE_DEBIT_CARD_NOT_PRINTED	<p>Scan usage arrived for a debit card that exists in the DebitCards table but was never printed, i.e. the DebitCards.Status column contains the value DEBIT_CARD_NOT_PRINTED.</p> <p><i>Conditions under which the test is performed:</i></p> <p>This test is performed for usages that are valid for guest admission and were not overridden by an operator, and for Void, Reversal and Adjustment usage.</p> <p><i>Possible causes of this status:</i></p> <p>Generally this error should only occur if a scan is performed in an offline state.</p>

#### 4.45 UsagePunches

The **UsagePunches** table is used for punching tickets with automated punching systems such as the Omron turnstile. This table stores punch positions for each use of a ticket (based on its access code). Only non-zero punch positions are stored.

##### Columns

Column	Type	Allow Nulls	Description
UsagePunchID	Int	N	Primary key, always unique.
AccessCode	Int	N	Foreign key to AccessCodes.AccessCode, specifying the access code this punch applies to.
Position	Int	Y	Use number (1-20).
UsagePunch	Int	Y	Punch position value corresponding to validated use number (1-255).

##### Indexes

Name	Kind	Columns	Purpose
PKUsagePunchesUsagePunchID	P	UsagePunchID	Primary Key
IXUsagePunchesAccessCode	A	AccessCode	Speed up central download of AccessCodes

## 4.46 ValidDates

A Valid Date record is a description of a date and time range for which a ticket is valid. It is used by TCON32 during the validation process of a ticket at an Access Control station. An Access Definition has a set of any number of Valid Date ranges. If there are no Valid Dates associated with an Access Definition, tickets with that definition's AccessCode are considered valid at any time (not including other restrictions). This table has been partially replaced by the **CalendarHeaders** and **CalendarDetails** tables, but **ValidDates** is still used by TCON32. In the future, this table will not be used.

### Columns

Column	Type	Allow Nulls	Description
AccessCode	Int	Y	Foreign key to AccessCodes.AccessCode, specifying the Access Definition to which this Valid Date record belongs.
Descr	Char(20)	Y	Description of the Valid Date Range.
FromDate	Int	Y	Used for determining starting date of validity, depending on FromDateKind. <sup>1</sup>
FromTime	Int	Y	Used for determining starting time of validity (on the starting date of validity), depending on FromTimeKind. <sup>2</sup>
ThruDate	Int	Y	Used for determining ending date of validity, depending on ThruDateKind. <sup>1</sup>
ThruTime	Int	Y	Used for determining ending time of validity (on the ending date of validity), depending on ThruTimeKind. <sup>2</sup>
FromDateKind	Char(2)	Y	Access Codes Valid Date Kind. <sup>1</sup>
FromTimeKind	Char(2)	Y	Access Codes Valid Time Kind. <sup>2</sup>
ThruDateKind	Char(2)	Y	Access Codes Valid Date Kind. <sup>1</sup>
ThruTimeKind	Char(2)	Y	Access Codes Valid Time Kind. <sup>2</sup>
DailyFrom	Int	Y	Starting time of validity (in HHMM format) on any date within FromDate and ThruDate.
DailyThru	Int	Y	Ending time of validity (in HHMM format) on any date within FromDate and ThruDate.
Days	Char(8)	Y	A set of seven Y/N characters indicating which days of the week that the ticket is valid. The first character represents Sunday. If none of the days are selected, the ticket will be valid every day of the week.
Based	Char(2)	Y	If using relative dates or times, this value indicates what date and/or time from which to base the new date and/or time. <sup>3</sup>
IsCounter	Bit	Y	This is TRUE on one record, to track the highest CounterID in use.
CounterID	Int	Y	Primary key, always unique.
Kind	Int	Y	Always 0. This column is not currently used by the system.
Master	Bit	Y	When selected, this rule must pass in order for ticket to be valid. When not selected, ticket can be valid even if this rule fails (as long as every rule with Master selected passes, and at least one rule with Master not selected passes).

### Indexes

(None)

#### <sup>1</sup> FromDateKind, ThruDateKind Values

Value	Gateway Constant Name	Description
A	IS_ABSOLUTE	Field described (FromDate or ThruDate) is an actual date in the format YYYYMMDD
R	IS_RELATIVE	Field described (FromDate or ThruDate) is relative to a base date, stored in the format YYMMDD, where YY, MM, and DD are the number of years, months, and days added to the base date
B	IS_BEFORE	Field described (FromDate or ThruDate) is relative before a base date, stored in the format YYMMDD, where YY, MM, and DD are the number of years, months, and days subtracted from the base date
M	IS_RELATIVE_MONTH	Field described (FromDate or ThruDate) is relative to a base date, similar to "R" (format YYMMDD), but using the "relative months" method of calculation

#### <sup>2</sup> FromTimeKind, ThruTimeKind Values

Value	Gateway Constant Name	Description
A	IS_ABSOLUTE	Field described (FromTime or ThruTime) is an actual time in the format HHMM
R	IS_RELATIVE	Field described (FromTime or ThruTime) is relative to a base time, stored in the format HHMM, where HH and MM are the number of hours and minutes added to the base time
B	IS_BEFORE	Field described (FromDate or ThruDate) is relative before a base time, stored in the format HHMM, where HH and MM are the number of hours and minutes subtracted from the base time

#### <sup>3</sup> Based Values

Value	Gateway Constant Name	Description
S	RELATIVE_TO_SALE	Relative dates and times are based on the date/time of sale
U	RELATIVE_TO_USE	Relative dates and times are based on the first ticket usage
E	RELATIVE_TO_EVENT	Relative dates and times are based on an event date/time
A	RELATIVE_TO_ARRIVAL	Relative dates and times are based on the order arrival date/time
T	RELATIVE_TO_TKTDATE	Relative dates and times are based on the ticket date
X	RELATIVE_TO_MEMBERSHIP_EXPIRATION	Relative dates and times are based on membership expiration

## 5 ACS Reservations

## 5.1 ACSReservations

The table storing the attributes of an ACS reservation.

### Columns

Column	Type	Allow Nulls	Description
ACSRreservationID	Int	N	Primary key, Always Unique
ConfirmationNumber	nvarchar(50)	N	The reservation confirmation number. Always unique.
VisualID	nvarchar(50)	N	Pass barcode.
ContactGUID	Uniqueidentifier	N	GUID associated to this contact. Used to uniquely identify this contact across any system.
PassID	Int	Y	PassID associated to this reservation, FK to Passes.PassNo
ReservedByContactGUID	Uniqueidentifier	N	GUID associated with the pass holder that made the reservation.
VisitDate	datetime	N	The date of the reservation.
Quantity	Int	N	The number of pass holders in the reservation.
AdmissionListID	Int	N	The admission header ID
Email	nvarchar(250)	N	Email address of the pass holder.
ReservationGroupID	Int	N	A unique identifier to group multiple pass holders to a single reservation.
Status	Int	N	The status of the reservation.
PassKindGroupID	Uniqueidentifier	N	Foreign Key reference to PasKindGroups table.
PassKindID	Int	N	Foreign Key reference to the pass kind.
ACSRreservationRuleSetID	Uniqueidentifier	N	Link to the AcsReservationRuleSets Table
CancellationDeadline	datetime	N	The final date in which a cancellation is allowed.
ModificationDeadline	datetime	N	The final date in which a modification is allowed.
CreatedAt	datetime	N	The date the reservation was made.
CancelledAt	datetime	N	The date the reservation was cancelled.
CancellationReason	nvarchar(max)	N	The narrative explaining the reason for the cancellation.
LastSequenceApplied	Int	N	Internal use. Incremented for each database event that occurs.
CancellationAllowed	bit	N	A flag to indicate if a cancellation is allowed.
ModificationAllowed	bit	N	A flag to indicate if a modification is allowed.
CreatedByGxUserGUID	Uniqueidentifier	N	The galaxy login username of the user making the reservation.
AvailabilityOverridden	bit	N	A flag to indicate if an availability issue has been overridden.
AvailabilityOverrideReason	nvarchar(max)	Y	The narrative explaining the reason for overriding the availability issue.
EligibilityOverridden	bit	N	A flag to indicate if an eligibility issue has been overridden.
EligibilityOverrideReason	nvarchar(max)	N	The narrative explaining the reason for overriding the eligibility issue.
ForgiveNoShowAt	datetime	N	The date the no-show infraction was forgiven.
ForgiveNoShowReason	nvarchar(max)	N	The narrative explaining the reason for forgiving a no-show infraction.
ACSRreservationGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique
RedeemedQuantity	Int	Y	The number of pass holders that redeemed this reservation
AvailabilityRequestID	nvarchar(50)	Y	The AvailabilityRequestID returned from the DLR availability API
ForgiveCancellationAt	datetime	Y	The date that a cancellation infraction was forgiven
ForgiveCancellationReason	nvarchar(max)	Y	The reason why a cancellation infraction was forgiven

### Indexes

Name	Kind	Columns	Purpose
PKACSRReservations	P	ACSRreservationID	Primary Key
IXACSRreservationGUID	A	ACSRreservationGUID	Alternate Key
IXConfirmationNumber		ConfirmationNumber	
IXACSRReservationsVisitDate		VisitDate,Status	
IXACSRReservationsVisualID		VisualID,Status	

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_STATUS_ACTIVE	Reservation is active.
1	ACS_RESERVATION_STATUS_CANCELLED	Reservation is cancelled.
2	ACS_RESERVATION_STATUS_USED	Reservation has been used.
3	ACS_RESERVATION_STATUS_NO_SHOW	Pass holder did not use the reservation.
4	ACS_RESERVATION_STATUS_SYSTEM_CANCELLED	Reservation is cancelled system wide or in bulk.
5	ACS_RESERVATION_STATUS_NO_SHOW_FORGIVE	A no-show infraction has been forgiven.
6	ACS_RESERVATION_STATUS_CANCELLED_FORGIVE	A cancelled reservation has been forgiven.

## 5.2 ACSReservationProfileConsequences

The table defining the consequences for reservation infractions.

### Columns

Column	Type	Allow Nulls	Description
ACSResProfConsequenceID	Int	N	Primary key, Always Unique
ACSRreservationProfileConsequenceGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique
VisualID	nvarchar(50)	N	Pass barcode.
CreatedAt	datetime	N	The date the consequence was made.
ConsequenceKind	Int	N	The indicator describing the relative timing of the consequence.
ActionKind	Int	N	The reservation status upon which the consequence is based.
EndDate	datetime	Y	The date the consequence will be lifted.
ThresholdLimit	Int	N	The threshold at which infractions will incur a consequence. See ActionKind for reservation status.
ThresholdDurationAmount	Int	N	The duration of the consequence.
ThresholdDurationKind	Int	N	The units applied to the duration of the consequence.
XParameter	Int	Y	A parameter needed for complex consequence rules. See ConsequenceKind for context.
YParameter	Int	Y	A parameter needed for complex consequence rules. See ConsequenceKind for context.
Expired	bit	N	A flag to indicate if a consequence is expired.
NotApplicable	bit	N	A flag to indicate if a consequence is applicable or not.
ShouldCreateNotification	Bit	N	A flag to indicate if an email should be sent regarding the consequence.
NotificationFrequencyKind	Int	Y	The frequency of the notification if ShouldCreateNotification is True.
ACSRreservationRuleSetConsequenceGUID	UniqueIdentifier	N	Link to the ACSReservationRuleSetConsequences table
EnforcedDate	datetime	Y	Date that the profile consequence was enforced
InfractionsIncurred	Int	N	Infractions counted against this particular profile consequence
CalendarYear	Int	N	The calendar year associated with infraction tracking for a given consequence.
CreateLiftedNotification	Bit	Y	A flag to indicate if an email should be sent when the consequence is lifted or expires.
LiftedNotificationFrequencyKind	Int	Y	The frequency of the notification if CreateLiftedNotification is True. (See NotificationFrequencyKind Values below.)

### Indexes

Name	Kind	Columns	Purpose
PKACSReservationProfileConsequences	P	ACSResProfConsequenceID	Primary Key

### <sup>1</sup> ThresholdDurationKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_DAY	The value for ThresholdDurationAmount will be interpreted as DAYS.
1	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_WEEK	The value for ThresholdDurationAmount will be interpreted as WEEKS.
2	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_MONTH	The value for ThresholdDurationAmount will be interpreted as MONTHS.
3	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_YEAR	The value for ThresholdDurationAmount will be interpreted as YEARS.
4	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_CALENDAR_YEAR	The consequence duration will last until the end of the current calendar year.

### <sup>2</sup> ActionKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_ACTION_KIND_VALID	The consequence applies to valid reservations.
1	ACS_RESERVATION_RULE_SET_ACTION_KIND_CANCELLATION	The consequence applies to canceled reservations.
2	ACS_RESERVATION_RULE_SET_ACTION_KIND_NO_SHOW	The consequence applies to no_show reservations.
3	ACS_RESERVATION_RULE_SET_ACTION_NEW	The consequence applies to new reservations.

### <sup>3</sup> ConsequenceKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_SUSPEND_FROM_OLDEST	Suspend new reservation access for X days from the oldest action.
1	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_SUSPEND_FROM_NEWEST	Suspend new reservation access for X days from the newest action.
2	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X_Y_FROM_OLDEST	Limit maximum reservations to X for Y days from oldest action.
3	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X_Y_FROM_NEWEST	Limit maximum reservations to X for Y days from newest action.
4	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X	Limit maximum active reservations to X.

### <sup>4</sup> NotificationFrequencyKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_BATCH	Notification happens in a daily batch.
1	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_REALTIME	Notification happens in real time.

### 5.3 ACSReservationProfiles

The table storing the information that determines a pass holder's ability to make reservations.

#### Columns

Column	Type	Allow Nulls	Description
ACSRreservationProfileID	Int	N	Primary key, Always Unique
VisualID	varchar(50)	N	Pass barcode.
ContactGUID	uniqueidentifier	N	GUID associated to this contact. Used to uniquely identify this contact across any system.
Status	Int	N	Indicates if a pass holder is blocked from making reservations.
BlockedReason	varchar(MAX)	Y	The reason why reservations are blocked.
ActiveReservations	Int	N	The number of unredeemed reservations.
AvailableReservations	Int	N	The number of reservations the pass holder may make.
CreatedAt	datetime	N	The date created.
NoShowConsequencesEndDate	datetime	Y	The date in which a no_show consequence will be lifted.
CancellationConsequenceEndDate	datetime	Y	The date in which a cancellation consequence will be lifted.
ValidConsequenceEndDate	datetime	Y	The date in which a valid consequence will be lifted.
NewConsequenceEndDate	datetime	Y	The date in which a new consequence will be lifted.
LastSequenceApplied	Int	N	Internal use. Incremented for each database event that occurs.
ACSRreservationProfileGUID	uniqueidentifier	N	Alternate Primary Key, Always Unique
BlockedReason	nvarchar(max)	Y	The reason for the block
NewConsequenceEndDate	datetime	Y	The date that the "new reservations" consequence will be lifted
PreviousVisualID	nvarchar(50)	Y	Pass barcode that was previously associated with this profile.

#### Indexes

Name	Kind	Columns	Purpose
PKACSRreservationProfiles	P	ACSRreservationProfileID	Primary Key
IXACSRreservationProfileGUID	A	ACSRreservationProfileGUID	Alternate Key
IXACSRreservationProfileContactGUID	F	ACSRreservationProfileContactGUID	Foreign Key
IXACSRreservationProfileVisualID		ACSRreservationProfileVisualID	

#### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	EL_STATUS_OK	The pass holder is not blocked from making reservations.
1	EL_STATUS_BLOCKED	The pass holder is blocked from making reservations.

#### 5.4 ACSReservationRuleSetAdmissionHeaders

Stores the link between ACS reservation rule sets and admission headers.

##### Columns

Column	Type	Allow Nulls	Description
ACSResRuleSetAdmissionHeaderID	Int	N	Primary key, Always Unique
ACSRReservationRuleSetAdmissionHeaderGUID	UniqueIdentifier	N	Table GUID
ACSRReservationRuleSetGUID	UniqueIdentifier	N	Link to the AcsReservationRuleSets Table
AdmissionHeaderID	Int	N	Link to AdmissionHeaders Table
IsDefault	Bit	N	True if this is the default admission header for the reservation rule set

##### Indexes

Name	Kind	Columns	Purpose
PKACSResRuleSetAdmissionHeaderID	P	ACSResRuleSetAdmissionHeaderID	Primary Key
IXACSRReservationRuleSetAdmissionHeaderGUID		ACSRReservationRuleSetAdmissionHeaderGUID	

## 5.5 ACSReservationRuleSetConsequences

The table defining the consequence parameters for reservation infractions.

### Columns

Column	Type	Allow Nulls	Description
ACSResRuleSetConsequenceID	Int	N	Primary key, Always Unique
ACSRreservationRuleSetGUID	UniqueIdentifier	N	Link to the AcsReservationRuleSets Table
ThresholdLimit	Int	N	The threshold at which infractions will incur a consequence. See ActionKind for reservation status.
ThresholdDurationAmount	Int	N	The duration of the consequence.
ThresholdDurationKind	Int	N	The units applied to the duration of the consequence.
ActionKind	Int	N	The reservation status upon which the consequence is based.
ConsequenceKind	Int	N	The indicator describing the relative timing of the consequence.
XParameter	Int	Y	A parameter needed for complex consequence rules. See ConsequenceKind for context.
YParameter	Int	Y	A parameter needed for complex consequence rules. See ConsequenceKind for context.
CreateNotification	Bit	N	A flag to indicate if an email should be sent regarding the consequence.
NotificationFrequencyKind	Int	Y	The frequency of the notification if CreateNotification is True.
ACSRruleSetConsequenceGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique
CreateInfractionNotification	Bit	Y	A flag to indicate if an email should be sent for each occurrence towards the total infraction.
AttributeValueGroupID	Int	Y	Links Items to the AttributeValues table.
CreateLiftedNotification	Bit	Y	A flag to indicate if an email should be sent when the consequence is lifted or expires.
LiftedNotificationFrequencyKind	Int	Y	The frequency of the notification if CreateLiftedNotification is True. (See NotificationFrequencyKind Values below.)

### Indexes

Name	Kind	Columns	Purpose
PKACSRruleSetConsequencesID	P	ACSRruleSetConsequenceID	Primary Key
IXACSRruleSetConsequencesGUID	A	ACSRreservationRuleSetConsequenceGUID	Alternate Key
IXACSRruleSetGUID	F	ACSRreservationRuleSetGUID	Foreign Key

### <sup>1</sup> ThresholdDurationKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_DAY	The value for ThresholdDurationAmount will be interpreted as DAYS.
1	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_WEEK	The value for ThresholdDurationAmount will be interpreted as WEEKS.
2	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_MONTH	The value for ThresholdDurationAmount will be interpreted as MONTHS.
3	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_YEAR	The value for ThresholdDurationAmount will be interpreted as YEARS.
4	ACS_RESERVATION_RULE_SET_CONSEQUENCE_THRESHOLD_DURATION_KIND_CALENDAR_YEAR	The consequence duration will last until the end of the current calendar year.

### <sup>2</sup> ActionKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_ACTION_KIND_VALID	The consequence applies to valid reservations.
1	ACS_RESERVATION_RULE_SET_ACTION_KIND_CANCELLATION	The consequence applies to canceled reservations.
2	ACS_RESERVATION_RULE_SET_ACTION_KIND_NO_SHOW	The consequence applies to no_show reservations.
3	ACS_RESERVATION_RULE_SET_ACTION_NEW	The consequence applies to new reservations.

### <sup>3</sup> ConsequenceKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_SUSPEND_FROM_OLDEST	Suspend new reservation access for X days from the oldest action.
1	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_SUSPEND_FROM_NEWEST	Suspend new reservation access for X days from the newest action.
2	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X_Y_FROM_OLDEST	Limit maximum reservations to X for Y days from oldest action.
3	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X_Y_FROM_NEWEST	Limit maximum reservations to X for Y days from newest action.
4	ACS_RESERVATION_RULE_SET_CONSEQUENCE_KIND_LIMIT_X	Limit maximum active reservations to X.

### <sup>4</sup> NotificationFrequencyKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_BATCH	Notification happens in a daily batch.
1	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_REALTIME	Notification happens in real time.

## 5.6 ACSReservationRuleSetNotifications

The table storing the parameters that define email triggers related to reservation activity.

### Columns

Column	Type	Allow Nulls	Description
ACSResRuleSetNotificationID	Int	N	Primary key, Always Unique
ACSRreservationRuleSetGUID	uniqueidentifier	N	Link to the AcsReservationRuleSets Table
ThresholdDays	Int	N	The number of days of advance notification.
ThresholdHours	Int	N	The number of hours of advance notification.
BeforeActionKind	Int	N	The indicator used to apply the threshold to the reservation event.
NotificationFrequencyKind	Int	N	The indicator used to determine the email frequency of the reservation event.
ACSRreservationRuleSetNotificationGUID	uniqueidentifier	N	Alternate Primary Key, Always Unique
AttributeValueGroupID	Int	Y	Links Items to the AttributeValues table.

### Indexes

Name	Kind	Columns	Purpose
PKACSRreservationRuleSetNotifications	P	ACSRreservationRuleSetNotificationID	Primary Key
IXACSRreservationRuleSetNotificationGUID	A	ACSRreservationRuleSetNotificationGUID	
IXACSRreservationRuleSetGUID	F	ACSRreservationRuleSetGUID	

### <sup>1</sup> BeforeActionKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_BEFORE_ACTION_KIND_CANCELLATION	Advance notification will be before the cancelation cutoff.
1	ACS_RESERVATION_RULE_SET_BEFORE_ACTION_KIND_RESERVATION	Advance notification will be before the reservation is redeemed.

### <sup>2</sup> NotificationFrequencyKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_BATCH	Notification happens in a daily batch.
1	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_REALTIME	Notification happens in real time.

## 5.7 ACSReservationRuleSets

The table storing the rules which govern the pass holder's ability to request, change, and cancel a reservation.

### Columns

Column	Type	Allow Nulls	Description
ACSRreservationRuleSetID	Int	N	Primary key, Always Unique
Name	nvarchar(50)	N	The name of the reservation rule set.
Description	nvarchar(250)	Y	The description of the reservation rule set.
MaximumActiveACSReservations	Int	N	Quantity of active reservations allowed.
RollingAvailabilityYears	Int	N	The number of years used to calculate the future date and time, relative to today, for which a pass holder may make a reservation.
RollingAvailabilityMonths	Int	N	The number of months used to calculate the future date and time, relative to today, for which a pass holder may make a reservation.
RollingAvailabilityDays	Int	N	The number of days used to calculate the future date and time, relative to today, for which a pass holder may make a reservation.
RollingAvailabilityHours	Int	N	The number of hours used to calculate the future date and time, relative to today, for which a pass holder may make a reservation.
AllowModifications	bit	N	A flag to indicate that this rule set allows a pass holder to make changes to a reservation.
ModificationCutoffYears	Int	N	The number of years used to calculate the date and time prior to a reservation for which a pass holder may change a reservation.
ModificationCutoffMonths	Int	N	The number of months used to calculate the date and time prior to a reservation for which a pass holder may change a reservation.
ModificationCutoffDays	Int	N	The number of days used to calculate the date and time prior to a reservation for which a pass holder may change a reservation.
ModificationCutoffHours	Int	N	The number of hours used to calculate the date and time prior to a reservation for which a pass holder may change a reservation.
AllowCancellations	bit	N	A flag to indicate that this rule set allows a pass holder to cancel a reservation.
CancellationCutoffYears	Int	N	The number of years used to calculate the date and time prior to a reservation for which a pass holder may cancel a reservation.
CancellationCutoffMonths	Int	N	The number of months used to calculate the date and time prior to a reservation for which a pass holder may cancel a reservation.
CancellationCutoffDays	Int	N	The number of days used to calculate the date and time prior to a reservation for which a pass holder may cancel a reservation.
CancellationCutoffHours	Int	N	The number of hours used to calculate the date and time prior to a reservation for which a pass holder may cancel a reservation.
ConsequenceEndDateKind	Int	N	End date of consequences. Actual end date is lesser of this value and frequency explicitly specified elsewhere in the rule set.
NotifyReservationCreated	bit	N	A flag to indicate an active email trigger for reservations created.
NotifyReservationCreatedFrequency	Int	Y	Daily batch or real-time.
NotifyReservationModified	bit	N	A flag to indicate an active email trigger for reservations changed.
NotifyReservationModifiedFrequency	Int	Y	Daily batch or real-time.
NotifyReservationCancelled	bit	N	A flag to indicate an active email trigger for reservations canceled.
NotifyReservationCancelledFrequency	Int	Y	Daily batch or real-time.
NotifyReservationNoShow	bit	N	A flag to indicate an active email trigger for no-show reservations.
NotifyReservationNoShowFrequency	Int	Y	Daily batch or real-time.
NotifyReservationEligibilityChanged	bit	N	A flag to indicate an active email trigger for changes in eligibility.
NotifyReservationEligibilityChangedFrequency	Int	Y	Daily batch or real-time.
NotifyReservationSystemCancelled	bit	N	A flag to indicate an active email trigger for system-initiated cancellations.
NotifyReservationSystemCancelledFrequency	Int	Y	Daily batch or real-time.
NotifyForgiveReservationNoShow	bit	N	A flag to indicate an active email trigger for forgiving no-show infractions.
NotifyForgiveReservationNoShowFrequency	Int	Y	Daily batch or real-time.
NotifyReservationUserCancelled	bit	N	A flag to indicate an active email trigger for user-initiated cancellations.
NotifyReservationUserCancelledFrequency	Int	Y	Daily batch or real-time.
PassKindGroupGUID	UniquedIdentifier	Y	Foreign Key to PassKindGroups table
Inactive	bit	N	A flag to indicate if the rule set is active.
ACSRreservationRuleSetGUID	UniquedIdentifier	N	Alternate Primary Key, Always Unique
AttributeValueGroupID	Int	Y	Links Items to the AttributeValues table.

### Indexes

Name	Kind	Columns	Purpose
PKACSRreservationRuleSetID	P	ACSRreservationRuleSetID	Primary Key
IXACSRreservationRuleSetGUID	A	ACSRreservationRuleSetGUID	Alternate Key

### <sup>1</sup> ConsequenceEndDateKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_CONSEQUENCE_END_DATE_KIND_NA	End Date is not applicable.
1	ACS_RESERVATION_RULE_SET_CONSEQUENCE_END_DATE_KIND_PASS_EXPIRATION	End Date is the pass expiration date.
2	ACS_RESERVATION_RULE_SET_CONSEQUENCE_END_DATE_KIND_END_OF_CALENDAR_YEAR	End Date is the end of the current calendar year.

**<sup>2</sup> Notify Frequency Values**

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_BATCH	Notification happens in a daily batch.
1	ACS_RESERVATION_RULE_SET_NOTIFICATION_FREQUENCY_KIND_REALTIME	Notification happens in real time.

## 5.8 ACSReservationRuleSetStatuses

The table storing the rules regarding a pass holder's eligibility to make a reservation based on the pass status.

### Columns

Column	Type	Allow Nulls	Description
ACSRreservationRuleSetStatusID	Int	N	Primary key, Always Unique
ACSRreservationRuleSetGUID	Uniqueidentifier	N	Link to the ACSReservationRuleSets Table
PassStatusKind	Int	N	The pass status.
EligibilityKind	Int	N	A flag to indicate if the pass holder is eligible to make a reservation.
ActionKind	Int	N	The action taken for active reservations given the current pass status.
XParameter	Int	N	The number of days applied to ActionKind CANCELED_X_DAYS_PRIOR.
ACSRreservationRuleSetStatusGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique

### Indexes

Name	Kind	Columns	Purpose
PKACSRreservationRuleSetStatuses	P	ACSRreservationRuleSetStatusID	Primary Key
IXACSRreservationRuleSetStatusGUID	A	ACSRreservationRuleSetStatusGUID	Alternate Primary Key
IXACSRreservationRuleSetGUID	F	ACSRreservationRuleSetGUID	Foreign Key

### <sup>1</sup> EligibilityKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_STATUSES_ELIGIBILITY_KIND_ELIGIBLE	The pass status allows a pass holder to make a reservation.
1	ACS_RESERVATION_RULE_SET_STATUSES_ELIGIBILITY_KIND_INELIGIBLE	The pass status does not allow a pass holder to make a reservation.
2	ACS_RESERVATION_RULE_SET_STATUSES_ELIGIBILITY_KIND_INELIGIBLE_CURRENT_YEAR	The pass status does not allow a pass holder to make a reservation in the current year.

### <sup>2</sup> ActionKind Values

Value	Gateway Constant Name	Description
0	ACS_RESERVATION_RULE_SET_STATUSES_ACTION_KIND_NO_ACTION	No action necessary.
1	ACS_RESERVATION_RULE_SET_STATUSES_ACTION_KIND_CANCELED_IMMEDIATELY	The reservation is to be canceled immediately.
2	ACS_RESERVATION_RULE_SET_STATUSES_ACTION_KIND_MOVED_TO_NEW_PASS	The reservation is to be moved to a new pass.
3	ACS_RESERVATION_RULE_SET_STATUSES_ACTION_KIND_CANCELED_X_DAYS_PRIOR	The reservation is to be canceled X days prior (see XParameter).
4	ACS_RESERVATION_RULE_SET_STATUSES_ACTION_KIND_CANCELED_CURRENT_YEAR_IMMEDIATELY	The reservation is to be canceled immediately if it is in the current year.

## 6 Authentication

## 6.1 AuthClaims

The table storing authentication claims associated with a resource.

### Columns

Column	Type	Allow Nulls	Description
AuthClaimID	Int	N	Primary Key, Always Unique
AuthResourceGUID	Uniqueidentifier	N	Foreign key to AuthResources
Type	nvarchar(250)	N	Name of claim that is added to auth token
AuthClaimGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique

## 6.2 AuthClientGrantTypes

The table storing grant types that a client can use to authenticate.

### Columns

Column	Type	Allow Nulls	Description
AuthClientGrantTypeID	Int	N	Primary Key, Always Unique
AuthClientGUID	UniqueIdentifier	N	Foreign key to AuthClients
GrantType	nvarchar(250)	N	Type of grant type that a client can use to authenticate
AuthClientGrantTypeGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique

### 6.3 AuthClients

The table storing details about clients that can authenticate.

#### Columns

Column	Type	Allow Nulls	Description
AuthClientID	Int	N	Primary Key, Always Unique
AbsoluteRefreshTokenLifetime	Int	N	Maximum lifetime of a refresh token in seconds. Defaults to 2592000 seconds / 30 days
AccessTokenLifetime	Int	N	Lifetime of access token in seconds (defaults to 3600 seconds / 1 hour)
AccessTokenType	Int	N	Specifies whether the access token is a reference token or a self contained JWT token (defaults to Jwt).
AllowAccessTokensViaBrowser	Bit	N	Specifies whether this client is allowed to receive access tokens via the browser
AllowOfflineAccess	Bit	N	Specifies whether this client can request refresh tokens (by requesting the offline_access scope)
AllowPlainTextPkce	Bit	N	Specifies whether clients using PKCE can use a plain text code challenge (not recommended - and default to false)
AllowRememberConsent	Bit	N	Specifies whether user can choose to store consent decisions. Defaults to true.
AlwaysIncludeUserClaimsInIdToken	Bit	N	When requesting both an id token and access token, should the user claims always be added to the id token instead of requiring the client to use the userinfo endpoint. Default is false.
AlwaysSendClientClaims	Bit	N	If set, the client claims will be sent for every flow. If not, only for client credentials flow (default is false)
AuthorizationCodeLifetime	Int	N	Lifetime of authorization code in seconds (defaults to 300 seconds / 5 minutes)
BackChannelLogoutSessionRequired	Bit	N	Specifies if the user's session id should be sent in the request to the BackChannelLogoutUri. Defaults to true.
BackChannelLogoutUri	nvarchar(2000)	Y	Specifies logout URI at client for HTTP based back-channel logout
ClientClaimsPrefix	nvarchar(200)	Y	If set, the prefix client claim types will be prefixed with. Defaults to client_. The intent is to make sure they don't accidentally collide with user claims.
ClientID	nvarchar(200)	N	Unique ID of the client
ClientName	nvarchar(200)	Y	Client display name (used for logging and consent screen)
ClientUri	nvarchar(2000)	Y	URI to further information about client (used on consent screen)
ConsentLifetime	Int	Y	Lifetime of a user consent in seconds. Defaults to null (no expiration).
Description	nvarchar(1000)	Y	Description of the client
EnableLocalLogin	Bit	N	Specifies if this client can use local accounts, or external IdPs only. Defaults to true.
Enabled	Bit	N	Specifies if client is enabled. Defaults to true.
FrontChannelLogoutSessionRequired	Bit	N	Specifies if the user's session id should be sent to the FrontChannelLogoutUri. Defaults to true.
FrontChannelLogoutUri	nvarchar(2000)	Y	Specifies logout URI at client for HTTP based front-channel logout
IdentityTokenLifetime	Int	N	Lifetime to identity token in seconds (defaults to 300 seconds / 5 minutes)
IncludeJwtId	Bit	N	Specifies whether JWT access tokens should have an embedded unique ID (via the jti claim).
LogoUri	nvarchar(max)	Y	URI to client logo (used on consent screen)
LogoutSessionRequired	Bit	N	Specifies if the user's session id should be sent to the LogoutUri. Defaults to true.
LogoutUri	nvarchar(2000)	Y	Specifies logout URI at client for HTTP based logout
PairWiseSubjectSalt	nvarchar(200)	Y	Salt value used in pair-wise subjectid generation for users of this client.
PrefixClientClaims	Bit	N	If set, all client claims will be prefixed with client_ to make sure they don't accidentally collide with user claims. Default is true.
ProtocolType	nvarchar(2000)	Y	Specifies the protocol that should be used for authentication.
RefreshTokenExpiration	Int	N	Expiration Type of the refresh token
RefreshTokenUsage	Int	N	Usage type of the refresh token
RequireClientSecret	Bit	N	Specifies whether this client needs a secret to request tokens from the token endpoint. Defaults to true
RequireConsent	Bit	N	Specifies whether a consent screen is required. Defaults to true.
RequirePkce	Bit	N	Specifies whether clients using an authorization code based grant type must send a proof key
SlidingRefreshTokenLifetime	Int	N	Sliding lifetime of a refresh token in seconds. Defaults to 1296000 seconds / 15 days
UpdateAccessTokenClaimsOnRefresh	Bit	N	Gets or sets a value indicating whether the access token (and its claims) should be updated on a refresh token request.
AuthClientGUID	Uniquedentifier	N	Alternate Primary Key, Always Unique

#### <sup>1</sup> RefreshTokenUsage Values

Value	Gateway Constant Name	Description
ReUse	ReUse	The refresh token handle will stay the same when refreshing tokens.
OneTime	OneTime	The refresh token handle will be updated when refreshing tokens. This is the default.

#### <sup>2</sup> RefreshTokenExpiration Values

Value	Gateway Constant Name	Description
Absolute	Absolute	The refresh token will expire on a fixed point in time (specified by the AbsoluteRefreshTokenLifetime).
Sliding	Sliding	When refreshing the token, the lifetime of the refresh token will be renewed (by the amount specified in SlidingRefreshTokenLifetime). The lifetime will not exceed AbsoluteRefreshTokenLifetime.

#### 6.4 AuthClientScopes

The table storing scopes for clients.

##### Columns

Column	Type	Allow Nulls	Description
AuthClientScopeID	Int	N	Primary Key, Always Unique
AuthClientGUID	Uniqueidentifier	N	Foreign Key to AuthClients
Scope	nvarchar(250)	N	Scope name of the allowed scope (Name from AuthScopes table)
AuthClientScopeGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique

## 6.5 AuthClientSecrets

The table storing secrets for clients.

### Columns

Column	Type	Allow Nulls	Description
AuthClientSecretID	Int	N	Primary Key, Always Unique
AuthClientGUID	Uniqueidentifier	N	Foreign Key to AuthClients
Description	nvarchar(2000)	Y	Description of secret
Expiration	datetime	Y	Expiration date of the secret
Type	nvarchar(250)	N	Type of secret
Value	nvarchar(2000)	Y	Value of secret
AuthClientSecretGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique

### <sup>1</sup> Type Values

Value	Gateway Constant Name	Description
SharedSecret	SharedSecret	Hashed secret
X509Thumbprint	X509Thumbprint	Thumbprint of X509 certificate
X509Name	X509Name	Name of X509 certificate
X509CertificateBase64	X509CertificateBase64	Base64 value of public key

## 6.6 AuthIdentityClaims

The table associating user claim types to an identity.

### Columns

Column	Type	Allow Nulls	Description
AuthIdentityClaimID	Int	N	Primary Key, Always Unique
AuthIdentityResourceGUID	UniqueIdentifier	N	Foreign Key to AuthIdentityResources
Type	nvarchar(250)	N	Type of claim. (From AuthClaims table)
AuthIdentityClaimGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique

## 6.7 AuthIdentityResources

The table associating user claim types to an identity.

### Columns

Column	Type	Allow Nulls	Description
AuthIdentityResourceID	Int	N	Primary Key, Always Unique
Description	nvarchar(2000)	Y	The value used on the consent screen.
DisplayName	nvarchar(250)	Y	The value used on the consent screen.
Emphasize	Bit	N	Specifies whether the consent screen will emphasize this scope. Defaults to false.
Enabled	Bit	N	Indicates if this resource is enabled and can be requested. Defaults to true.
Name	nvarchar(250)	N	The unique name of the identity resource. This is the value a client will use for the scope parameter in the authorize request.
Required	Bit	N	Specifies whether the user can de-select the scope on the consent screen (if the consent screen wants to implement such a feature). Defaults to false.
ShowInDiscoveryDocument	Bit	N	Specifies whether this scope is shown in the discovery document. Defaults to true.
AuthIdentityResourceGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique

## 6.8 AuthPersistedGrants

The table that represents the tokens created as they are granted and updated when they are revoked.

### Columns

Column	Type	Allow Nulls	Description
Key	nvarchar(200)	N	Primary Key, Always Unique
ClientId	nvarchar(200)	N	ClientId that is associated with the grant (ClientId from AuthClients table)
CreationTime	datetime	N	Date and time that the grant was generated.
Data	nvarchar(max)	N	JSON Data describing the grant.
Expiration	datetime	N	Date and time that the grant will expire.
SubjectId	nvarchar(200)	Y	"sub" claim of the token
Type	nvarchar(50)	N	Type of grant token

## 6.9 AuthResources

The table that represents the tokens created as they are granted and updated when they are revoked.

### Columns

Column	Type	Allow Nulls	Description
AuthResourceID	Int	N	Primary Key, Always Unique
Description	nvarchar(2000)	Y	Value shown on the consent screen
DisplayName	nvarchar(250)	Y	Value shown on the consent screen
Enabled	Bit	N	Indicates if this resource is enabled and can be requested. Defaults to true.
Name	nvarchar(250)	N	The unique name of the API. This value is used for authentication with introspection and will be added to the audience of the outgoing access token.
AuthResourceGUID	UniqueIdentifier	N	Alternate Primary Key, Always Unique

## 6.10 AuthScopes

The table that represents the tokens created as they are granted and updated when they are revoked.

### Columns

Column	Type	Allow Nulls	Description
AuthScopeID	Int	N	Primary Key, Always Unique
AuthResourceGUID	Uniqueidentifier	N	Foreign Key to AuthResources
Description	nvarchar(2000)	Y	Value shown on the consent screen
DisplayName	nvarchar(250)	Y	Value shown on the consent screen
Emphasize	Bit	N	Specifies whether the consent screen will emphasize this scope. Defaults to false.
Name	nvarchar(250)	Y	The unique name of the scope. This is the value a client will use for the scope parameter in the authorize/token request.
Required	Bit	N	Specifies whether the user can de-select the scope on the consent screen. Defaults to false.
ShowInDiscoveryDocument	Bit	N	Specifies whether this scope is shown in the discovery document. Defaults to true.
AuthScopeGUID	Uniqueidentifier	N	Alternate Primary Key, Always Unique

## 7 Blacklist

Provides a list of keys that can be restricted from performing actions in Galaxy. For example a user could create a list of customers that cannot purchase tickets based on their government ID.

## 7.1 Blacklists

This table stores the definition of the blacklist “rules”.

### Columns

Column	Type	Allow Nulls	Description
BlacklistId	Int	No	Primary key.
Name	Text	No	A user defined name for the blacklist “rule”.
Format	Text	No	The format, or “mask” that’s used to create the unique blacklisted string / key.
EncryptKey	Int	No	Does the key as defined by this format need to be encrypted.

### Indexes

Name	Kind	Columns	Purpose
PKBlacklistId	P	BlacklistId	Primary Key.

## 7.2 BlacklistMembers

Stores the actual blacklist records, the instances of each blacklist entry that defined by the rule in the definition.

### Columns

Column	Type	Allow Nulls	Description
BlacklistMemberId	Int	No	Primary key.
BlacklistId	Int	No	A link to the definition that this entry was created from.
BlacklistKey	Text	No	The unique blacklist key that was generated from the format defined by the header definition.
GxKeyId	Int	No	Foreign key to GxKeys.GxKeyId, this element defines what GxKeyId to use to decrypt the BlacklistKey. If this value is non-zero, the system assumes the text in the Endorsement element is encrypted.

### Indexes

Name	Kind	Columns	Purpose
PKBlacklistMemberId	P	BlacklistMemberId	Primary Key.

### 7.3 BlacklistActions

Stores any overridden values that differ from what the definition dedicates as the defaults of the definition.

#### Columns

Column	Type	Allow Nulls	Description
BlacklistActionId	Int	No	Primary key.
BlacklistMemberId	Int	No	A link to the BlacklistMembers table
ActionType	Int	No	An enumeration that will represent the function.

#### Indexes

Name	Kind	Columns	Purpose
PKBlacklistActionId	P	BlacklistActionId	Primary Key.

#### 7.4 BlacklistActionTypeNames

This table is not user editable, it's just a lookup table with the function enumerations.

##### Columns

Column	Type	Allow Nulls	Description
ActionType	Int	No	Primary key.
Name	Int	No	Name of the action.

##### Indexes

Name	Kind	Columns	Purpose
PKActionType	P	ActionType	Primary Key.

## 7.5 BlacklistDefaultActions

This table Stores the default functions and the associated actions.

### Columns

Column	Type	Allow Nulls	Description
BlacklistDefaultActionId	Int	No	Primary key.
BlacklistId	Int	No	Name of this rule.
ActionType	Int	No	An enumeration that will represent the function .

### Indexes

Name	Kind	Columns	Purpose
PKBlacklistDefaultActionId	P	BlacklistDefaultActionId	Primary Key.

## 8 Debit

Galaxy Debit module provides a currency-less method of payment for anything that can be purchased at a Galaxy POS station. The primary benefit is the convenience to guests, which results in stimulated spending. By offering a debit system, guests can exchange cash for an easy to carry or durable debit card that can be used throughout the attraction to make purchases. This is particularly useful in water parks where paper currencies can become soaked and unusable. With the Galaxy debit system, the attraction can now hold the guests money for them, making it more convenient for spending throughout the day and during return visits.

## 8.1 DebitTypes

This table stores debit card type definitions used by the debit module. Each debit card created by the system references exactly one debit type. A debit type is used to define rules for a grouping of debit cards.

### Columns

Column	Type	Allow Nulls	Description
DebitTypeID	Int	N	Primary key, always unique.
MediaID	Int	Y	Foreign key reference to Media.MediaID.
Description	Char(40)	Y	A text description of this debit type row.
ExpirationType	Int	Y	Used to determine how a debit card expires <sup>1</sup> .
ExpirationDate	DateTime	Y	The day debit cards with this type expire. Only used when ExpirationType = 0.
ExpirationDays	Int	Y	The number of days after the initial sale that debit cards with this type expire. Only used when ExpirationType = 1.
JournalizeLookups	Bit	Y	Set to true when the system should journalize occurrences of a SV/Debit card lookup.
ExternalHost <sup>2</sup>	Int	Y	Specifies the data connection that is used for operations (activation, recharge, etc) that involved Stored Value cards of this type.
UseExternalMedia	Bit	Y	Set to true if external media will be used for the stored value instead of media generated within Galaxy. This is used only for the Gateway Stored Value system. The external media ID will be used as the visual ID instead of a unique ID generated by the system.
PluginID	Int	Y	Foreign key to Plugins.PluginID. Indicates the plugin to use as the host for this debit type. This is used in place of the ExternalHost field when a plugin is used for stored value.
UnloadPLU	NChar(20)	Y	Foreign key to Items.PLU. Indicates the PLU of the item that is added to the transaction when an unload is performed with this debit type. This overrides the unload PLU from general configuration. If no unload PLU is specified on the debit type, the unload PLU from general configuration will be used.

### Indexes

Name	Kind	Columns	Purpose
PKDebitTypesDebitTypeID	P	DebitTypeID	Primary Key.

### <sup>1</sup> ExpirationType Values

Value	Gateway Constant Name	Description
0	DEBIT_EXPIRE_DATE	When ExpirationType is one, the ExpirationDate value will dictate when cards with this type expire.
1	DEBIT_EXPIRE_DAYS	When ExpirationType is two, the ExpirationDays value will dictate when cards with this type expire.

### <sup>2</sup> ExternalHost Values

Value	Gateway Constant Name	Description
0	SV_GTS_HOST	Stored value operations are routed to the internal Gateway stored value processor.
1	SV_FLEXCACHE_HOST	Stored value operations are routed to the FlexCache stored value processor.
2	SV_STRATUS_RC_HOST	Stored value operations are routed to the Stratus RC stored value processor.
3	SV_STRATUS_GT_HOST	Stored value operations are routed to the Stratus GT stored value processor.
4	SV_WILDCARD_HOST	Stored value operations are routed to the WCS stored value processor
5	SV_SKIDATA_SC_HOST	Stored value operations are routed to the Skidata smart cards stored value processor
6	SV_NECTAR_HOST	Stored value operations are routed to the Nectar stored value processor
7	SV_EFSNET_HOST	Stored value operations are routed to the FDMS stored value processor
8	SV_PDC_HOST	Stored value operations are routed to the PDC stored value processor
9	SV_VALUE_LINK_HOST	Stored value operations are routed to the ValueLink stored value processor
8	SV_PLUGIN_HOST	Stored value operations are processed through a plugin

## 8.2 DebitCards

The DebitCards table contains all debit cards sold by the system and the current state of each card (status, when it expires, how much value remains on the card, etc).

### Columns

Column	Type	Allow Nulls	Description
DebitCardID	Int	N	Primary key, always unique.
VisualID	VarChar(60)	N	Alternate key, always unique. The ID number for the debit card that the system uses when retrieving a row.
DebitTypeID	Int	Y	Foreign key reference to DebitTypes.DebitTypeID.
Status	Int	Y	The current status of the debit card <sup>1</sup> .
ExpirationDate	DateTime	Y	The date the debit card expires. NULL indicates that the card does not expire.
Balance	Money	Y	The remaining value on the debit card.
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the debit card number (VisualID).
OrderID	Int	Y	The order the debit card was issued from. Zero indicates that the debit card was issued from a POS and was not part of an order.
ContactID	Int	Y	The contact associated to this stored value account. This column is populated whenever a contact is present in the transaction where the stored value card is purchased.

### Indexes

Name	Kind	Columns	Purpose
PKDebitCardsDebitCardID	P	DebitCardID	Primary Key.
AKDebitCardsVisualID	A	VisualID	Alternate key, insures uniqueness. Optimizes debit card lookups performed by the system.
IXDebitCardsContactID	F	ContactID	Foreign key to CustContacts.CustContactID

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	DEBIT_CARD_VALID	Valid debit card.
1	DEBIT_CARD_VOIDED	The debit card has been voided and is no longer valid.
2	DEBIT_CARD_RETURNED	The debit card has been returned and is no longer valid.
3	DEBIT_CARD_EXPIRED	The debit card has expired and is no longer valid.
4	DEBIT_CARD_INACTIVE	The debit card is inactive and is not valid.
5	DEBIT_CARD_NOT_PRINTED	The debit card has been issued but has not been printed.
6	DEBIT_CARD_REPRINTED	The debit card has been reprinted and is no longer valid.
7	DEBIT_CARD_UPGRADED	The debit card has been upgraded (in a package) and is no longer valid.

### 8.3 DebitCardLog

This table is used to store a history of debit card activity.

#### Columns

Column	Type	Allow Nulls	Description
DebitCardLogID	Int	N	Primary key, always unique.
TransDate	DateTime	Y	The date and time this activity occurred.
TransNo	Int	Y	The POS transaction number this activity occurred in.
TransLineNo	Int	Y	The value for this column is obtained from the journal Tkt/Tran value (Tickets.TktIndex). Its only function is to uniquely reference a detail in the journal based on the information in any given row (in this table). The debit module uses it to prevent double applying recharges in the event the database sender pointer is reset.
NodeNo	Int	Y	The node number this activity occurred at. TransNo + NodeNo can be used to uniquely identify transaction in JnlHeaders.
CardLogType	Int	Y	The type of activity. <sup>1</sup>
VisualID	VarChar(60)	Y	The VisualID of the debit card this activity is for. Foreign key reference to DebitCards.VisualID.
AuxID	VarChar(40)	Y	An auxiliary ID number on used for certain types of activity. Currently, it's only used when CardLogType = 2 <sup>2</sup>
PreviousBalance	Money	Y	The balance on the debit card before this activity occurred.
CurrentBalance	Money	Y	The balance on the debit card before this activity occurred.
Status	Int	Y	0 or 1 (0 = Processed, 1 = Not Processed) <sup>3</sup>
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the debit card number (VisualID).
ExternalTransNo	Int	Y	The unique transaction number of the debit transaction from the external system
ExternalStation	VarChar(30)	Y	The node/station number (or name) from the external system that the debit transaction occurred at
SourceID	Int	Y	Foreign key to eGalaxySources.eGalaxySourceID. This is the source system that sent the debit transaction.

#### Indexes

Name	Kind	Columns	Purpose
PKDebitCardLogDebitCardLogID	PK	DebitCardLogID	Primary Key.
IXDebitCardLogVisualID	IX	VisualID	Index used to optimize history queries for a given debit card.
IXDebitCardLogSVBalance	IX	CardLogType, PreviousBalance, CurrentBalance, TransDate	Speed up SV Reports
IXDebitCardLogSourceID	IX	SourceID	For searches on foreign key 'SourceID'
IXDebitCardLogTransNoNodeNo	IX	TransNo, NodeNo	For JournalSender plug in

#### <sup>1</sup> CardLogType Values

Value	Gateway Constant Name	Description
0	CARD_LOG_SALE	This entry is for a debit card sale.
1	CARD_LOG_RECHARGE	This entry is for a debit card recharge. This type activity will typically increase the balance.
2	CARD_LOG_CHARGE	This entry is for a debit card used to purchase goods. (When a transaction is "charged" to a debit card, i.e., a card is debited). This type activity will decrease the balance.
3	CARD_LOG_CREDIT	This entry is used when goods purchased in a previous transaction are returned and the value of the transaction is credited to a debit card. This type of activity will increase the balance.
4	CARD_LOG_RETURN	This entry is used when a debit card is returned. The balance will not be affected by a return.
5	CARD_LOG_VOID	This entry is used when a debit card is voided. The balance will not be affected by a void.
6	CARD_LOG_UNLOAD	This entry is for a debit card unload. This type of activity will zero the balance.
7	CARD_LOG_RECHARGE_VOID	This entry is used when a debit card's recharge is voided.
8	CARD_LOG_ERASE	Debit card usage log entry made when a card is erased
9	CARD_LOG_REPAIR	Debit card usage log entry made when a card is repaired
10	CARD_LOG_OVERRIDE	Debit card usage log entry made when a card has a negative balance during unload and the user overrides the owed balance.
11	CARD_LOG_LOCKER_ADJUST	This entry is added by eGalaxy when the locker transaction total received by eGalaxy in the ImportDebitTransaction message does not match the amount from the unprocessed locker transactions in the DebitCardLog table. This transaction is added to account for the difference so the amount on the RFID chip matches the amounts in SQL.
12	CARD_LOG_DEACTIVATE	This entry is used when an account is deactivated.

#### <sup>2</sup> AuxID Values - only when CardLogType = 2

Value	Gateway Constant Name	Description
0	DEBIT_AUTH_APPROVED	A debit authorization was successful.
1	DEBIT_AUTH_NO_BALANCE	A debit authorization was declined due insufficient funds on the card.
2	DEBIT_AUTH_EXPIRED	A debit authorization was declined because the card was expired.
3	DEBIT_AUTH_RETURNED	A debit authorization was declined because the card was already returned.
4	DEBIT_AUTH_VOIDED	A debit authorization was declined because the card was already voided.
5	DEBIT_AUTH_NOT_FOUND	A debit authorization was declined because the card did not exist.
6	DEBIT_AUTH_ALREADY_RECHARGED	This entry is used when a user attempted to recharge a debit card and use that same debit card to tender the transaction. This is not allowed since the action would essentially add and remove the same amount from the debit card in that transaction.

#### <sup>3</sup> Status Values

Value	Gateway Constant Name	Description
0	DEBITCARDLOG_STATUS_PROCESSED	Processed Transaction
1	DEBITCARDLOG_STATUS_UNPROCESSED	Unprocessed Transaction

#### 8.4 DebitLockouts

The DebitLockouts table contains locked out Debit cards by VisualID, including a Reason for the lockout.

##### Columns

Column	Type	Allow Nulls	Description
DebitLockoutID	Int	N	Primary key, always unique
VisualID	VarChar(40)	N	VisualID of Card
Reason	VarChar(255)	Y	Reason for lockout

##### Indexes

Name	Kind	Columns	Purpose
PKDebitLockoutsDebitLockoutID	P	DebitLockoutID	Primary Key.
IXDebitLockoutsVisualID	Unique	VisualID	Unique VisualID

## 8.5 SVDataConnectionValues

This table contains configuration settings used for certain types of Stored Value Data Connections.

### Columns

Column	Type	Allow Nulls	Description
SVDataConnectionValueID	Int	N	Primary key, always unique
DataConnection	Int	N	The type of data connection the option applies to <sup>1</sup>
DataName	VarChar(40)	N	The name (possibly abbreviated) of the option, used by the system to look up the setting (in conjunction with DataConnection)
DataProvider	VarChar(128)	Y	The configured value for the option

### Indexes

Name	Kind	Columns	Purpose
PKSVDatavalue	P	SVDataConnectionValueID	Primary Key

### <sup>1</sup> DataConnection Values

Value	Gateway Constant Name	Description
0*	SV_GTS_HOST	Gateway Stored Value System
1*	SV_FLEXCACHE_HOST	FlexCache
2*	SV_STRATUS_REWARDS_HOST	Stratus - RC
3*	SV_STRATUS_GIFT_HOST	Stratus - GC
4	SV_WILDCARD_HOST	WCS - GC
5	SV_SKIDATA_SC_HOST	
6	SV_NECTAR_HOST	
7	SV_EFSNET_HOST	
8	SV_PDC_HOST	
9	SV_VALUE_LINK_HOST	

\*Note: These types of data connections do not currently use configuration options stored in this table.

## 9 eGalaxy

eGalaxy is the internet ticketing and e-commerce module for Galaxy Revenue Management.

The eGalaxy solution supplies one of the "channels" supported by Galaxy for multi-channel sales. Specifically, the Galaxy system administrator creates products (items) and determines the sales channels through which they are sold. Sales channels can include front gate ticket booths, self-service ticketing kiosks, and an internet web site.

An eGalaxy system consists of two modules: Web Store and eGalaxy Server software.

eGalaxy Server provides the communications interface between a Galaxy system and a web store.

## 9.1 AppendedOrderLines

When eGalaxy Server processes an "Append" XML message, it creates a record in AppendedOrderLines for each "appended" order line that needs to be processed by Web Order Processor. When Web Order Processor begins processing an OrderCommand, it checks the AppendedOrderLines table to determine if the OrderCommand represents an "Append" message and, if so, which order lines were appended to the order.

### Columns

Column	Type	Allow Nulls	Description
AppendedOrderLineID	Integer	N	Primary key, gateway counter
OrderLineID	Integer	N	FK reference to the OrderLines table
OrderCommandID	Integer	N	FK reference to the OrderCommands table
IsIssued	bit	N	If this order line was issued by eGalaxy Server, contains the value 1. If this order line was not issued by eGalaxy Server, contains 0.

### Indexes

Name	Kind	Columns	Purpose
PKAppendedOrderLineID	P	AppendedOrderLineID	Primary Key, gateway counter

## 9.2 CustomerGroups

**CustomerGroups** table contains a group of the customers. The table is used by eGalaxy to validate the customer and order information before importing the order into Order Entry. The contents of the CustomerGroups table can be found in the **CustomerGroupDetails** table.

### Columns

Column	Type	Allow Nulls	Description
CustomerGroupID	Integer	N	Primary key. Obtained from GatewayCounters.
GroupDescr	VarChar(80)	N	Text description of group.

### Indexes

Name	Kind	Columns	Purpose
PKCustomerGroupsCustGroupID	P	CustomerGroupID	Primary key.

### 9.3 CustomerGroupDetails

**CustomerGroupDetails** table contains the details for the **CustomerGroups** table. A CustomerGroup is a collection of customers. This table is used by eGalaxy while importing an order into Order Entry.

#### Columns

Column	Type	Allow Nulls	Description
CustomerGroupDetailID	Integer	N	Primary key. Obtained from GatewayCounters.
CustomerGroupID	Integer	N	FK reference to CustomerGroup.CustomerGroupID
CustomerID	Integer	N	FK reference to Customers
eGalaxyPaidOrders	Bit	N	Set to require payment to be submitted with all eGalaxy orders (even if customer has billing account)

#### Indexes

Name	Kind	Columns	Purpose
PKCustGrpDetailsCustGrpDtlID	P	CustomerGroupDetailID	Primary key.
AKCustGrpDetailsGrpIDCustID	A	CustomerGroupDetailID, CustomerGroupID, CustomerID	Alternate key.
IXCustomerGroupDtlsCustGroupID		CustomerGroupID	Index necessary to speed up the query which returns all details for the given CustomerGroupID

## 9.4 eGalaxyConfig

The eGalaxyConfigs table is used to save the configurations specific to an eGalaxy Import node. Each record in the table contains different configuration options for an eGalaxy Import node.

### Columns

Column	Type	Allow Nulls	Description
eGalaxyConfigID	Int	N	Primary key, always unique
NodeNo	Int	N	Node number, always unique
AgencyNo	Int	N	Agency in which the current node is running
CompanyID	Int	N	Company of the current node
UserID	Int	N	UserID under which all transaction are journalized
CreditFOP	Int	N	FOP that will be used as charge FOP
ImportFromWebsite	Bit	Y	Get new orders from a website - Default 0 - False
WebURL	Varchar(128)	Y	URL of the website
WebUsername	Varchar(128)	Y	Username to log on to the website if needed
WebPassword	Varchar(128)	Y	Encrypted Password to log on to the website if needed
ImportFromFile	Bit	Y	Get new orders from a file - Default 0 - False
InputDir	Varchar(256)	Y	Directory to monitor for new orders, when ImportFromFile is TRUE
ErrorDir	Varchar(256)	Y	Directory to save error files, when ImportFromFile is TRUE
ArchiveDir	Varchar(256)	Y	Directory to store files after processing is done, when ImportFromFile is TRUE
WebAutoCheckInterval	Int	Y	Interval in seconds to auto check for new orders from a website
FileAutoCheckInterval	Int	Y	Interval in seconds to auto check for new orders from a file
LastTranNo	Int	Y	Latest Transaction Number in use for the current node
AppUsername	Varchar(40)	Y	Logon Username used by eGalaxy to connect to a remote website application to check for any pending orders to download, and send reply back to it with a status of the order.
AppPassword	Varchar(40)	Y	Encrypted Password for AppUserName.
EmailName	NVarChar(Max)	Y	Email name of the client who will send a summary status of the imported files or orders downloaded from the website.
EmailAddress	NVarChar(Max)	Y	Email address of the client who will send a summary status of the imported files or orders downloaded from the website.
ReplyToEmailAddress	NVarChar(Max)	Y	Reply-to field on email will be set to this address
EndOfDay	Integer	Y	End of day time, used when sending email for the entire day's activity to an eGalaxy source. Format: hhmm in 24 hour format.
DownloadSurveys	Bit	Y	to down load surveys from Web store (just like orders)
DefaultEmailName	NVarChar(Max)	Y	Email Name of the who will receive the e-mail
DefaultEmailAddress	NVarChar(Max)	Y	Email Address of the who will receive the e-mail
ErrorEmailTemplate	NVarChar(Max)	Y	Template to use as a default for the Error Notification E-mail
ReleaseIdleWebSessions	Bit	Y	If set to 1, system finds and releases the idle Web sessions
SessionInactivityTime	Int	Y	Inactivity time period (in minutes) before a session is considered idle
SessionReleaseAutoCheckInterval	Int	Y	Interval in minutes to auto check for idle sessions in eGalaxySessions table
PurgeClosedSessionsAtEOD	Bit	Y	If set to 1, system deletes the closed sessions from eGalaxySessions table during End of day process
PurgeClosedSessionDays	Int	Y	Number of days to wait before deleting a closed sessions from eGalaxySessions table
OrderPickupCount	Int	Y	The number of orders to download in a single PickupOrders request (default = 5).
SurveyPickupCount	Int	Y	The number of surveys to download in a single PickupSurveys request (default = 5).
OrderPickupErrorNotifyInterval	Int	Y	Time interval in minutes to wait before re-sending the error notification for unsuccessful order download

### Indexes

Name	Kind	Columns	Purpose
PkeGalaxyConfigeGalaxyConfigID	P	eGalaxyConfigID	Primary Key.
AkeGalaxyConfigNodeNo	A	NodeNo	Unique Key

## 9.5 eGalaxyErrorLog

This table contains errors generated while importing an order by eGalaxy.

### Columns

Column	Type	Allow Nulls	Description
eGalaxyErrorLogID	Int	N	Primary key, always unique
Reference	Char(128)	N	<OrderID> element of an eGalaxy XML order.
eGalaxySourceID	Int	N	Foreign Key to eGalaxySources.eGalaxySourceID
ErrorDate	DateTime	N	Working date of error
ErrorType	Int	N	Type of error <sup>1</sup>
ErrorCode	Int	N	Error Code.
ErrorText	Varchar(256)	N	Description of ErrorCode
Command	Varchar (20)	Y	Command of the request that generated this error e.g. For a ticket activation error this field contains "Activate" or "Cancel". For an order request this field contains "Add", "Modify", or "Cancel"

### Indexes

Name	Kind	Columns	Purpose
PkeGErrorLogErrorLogID	P	eGalaxyErrorLogID	Primary Key.
IXeGErrLogeGalaxySrcIDErrDate		eGalaxySourceID, ErrorDate	Index to speed up the loading of errors for a given working date for eGalaxy End Of Day processing.
IXeGErrLogReferenceErrCode		Reference, ErrorCode	Index to speed up the loading of errors for a given order related to SetOrderStatus timeouts.

<sup>1</sup> ErrorType Values

Value	Gateway Constant Name	Description
0	etUnknown	Errors generated but unknown to eGalaxy
1	etInternal	Errors internal to eGalaxy, i.e. Exception.
2	etFormat	XML order is not in a valid form. i.e. missing <order> element
3	etFile	Importing file has a problem
4	etOrderLevel	Error occurred while validating XML order
5	etAuthentication	User authentication error
6	etMessage	XML error during realtime processing
7	etEvent	Event related errors
8	etTicketActivation	Ticket Activation related errors.
9	etQueryTicket	Ticket Query related errors

## 9.6 eGalaxyErrorSeverities

This table contains a list of error codes for each eGalaxy source that are configured to be processed as a warning rather than as an error. The system only allows certain error codes to be processed as warnings (for many errors, it is not possible to process an order and open it in Order Entry if the error occurred).

### Columns

Column	Type	Allow Nulls	Description
eGalaxyErrorSeverityID	Int	N	Primary key, always unique
eGalaxySourceID	Int	N	FK to eGalaxySources.eGalaxySourceID
ErrorCode <sup>1</sup>	Int	N	Code to be defined as an error or warning
Severity <sup>2</sup>	Int	N	Indicates whether the potential problem should be handled as an error or as a warning

### Indexes

Name	Kind	Columns	Purpose
PkeGalaxyErrorSeverityID	P	eGalaxyErrorSeverityID	Primary Key.

### <sup>1</sup> ErrorCode Values

Value	Gateway Constant Name	Description
105	INVALID_PRICE_ERROR	Ticket/Item price does not match Galaxy's price for the same item
114	INVALID_VISITDATE_ERROR	VisitDate is either out of range, or has an invalid value
123	eGALAXY_SOURCE_INACTIVE_ERROR	Source is inactive
140	ORD_PASS_NOT_RENEWABLE	The pass is not renewable
144	ORD_ZERO_PROMOTION_CODE	Promotion code used in the order has no uses left
145	ORD_EVENT_IS_NOT_ON_SALE	Event is not on sale
146	ORD_EVENT_DOES_NOT_HAVE_ENOUGH_CAPACITY	Event does not have enough quantity available

### <sup>2</sup> Severity Values

Value	Gateway Constant Name	Description
0	esError	Condition is treated as an error, order will be rejected when this occurs. This is the default value used when there is no entry for an error code in this table; the system does not currently save records having this value.
1	esWarning	Condition is treated as a warning, order will still be processed.

## 9.7 eGalaxyJournalFiscalSummaryQueue

This table maintains a queue of Reseller nodes that eGalaxy will generate journal fiscal summaries for during end of day processing. Nodes are automatically added to this table when the CC\_CODE\_JOURNALIZE\_FISCAL\_SUMMARIES (1083) option is enabled and an eGalaxy CreateTransaction request is processed for a Reseller node.

### Columns

Column	Type	Allow Nulls	Description
eGalaxyJnlFiscalSummaryQueueID	Int	N	Primary key, always unique, Identity.
NodeNo	Int	N	Node number, always unique
LastReportDate	DateTime	Y	The last date that the journal fiscal summaries were generated for the node.
SourceNodeNo	Int	N	The node that is currently processing the record. This is used to prevent concurrent processing of a single node.

### Indexes

Name	Kind	Columns	Purpose
PKeGalaxyJnlFiscalSummaryQueueID	P	eGalaxyJnlFiscalSummaryQueueID	Primary Key.
AKeGalaxyJournalFiscalSummaryQueueNodeNo	A	NodeNo	Unique Key

## 9.8 eGalaxyPasswordResetTokens

The eGalaxyPasswordResetTokens table is used to store requests from guests of the eGalaxy Web Store to reset their password.

### Columns

Column	Type	Allow Nulls	Description
eGalaxyPasswordResetTokenID	Int	N	Primary key, always unique
eGalaxyPasswordResetTokenGUID	UniqueIdentifier	N	Password reset token GUID, provided to the guest via URL to associate to this password reset request
SystemLogonID	Int	N	Foreign key to the SystemLogons table
TokenDate	DateTime	N	Date and time of the password reset request, always stored as UTC

### Indexes

Name	Kind	Columns	Purpose
PKeGalaxyPasswordResetTokenID	P	eGalaxyPasswordResetTokenID	Primary Key
IXeGalaxyPasswordResetTokenGUID	A	eGalaxyPasswordResetTokenGUID	Unique Key
IXTokenDate	A	TokenDate	Non-unique Key
IXSystemLogonID	A	SystemLogonID	Non-unique Key

## 9.9 eGalaxyQueryTicketFieldRestrictions

The eGalaxyQueryTicketFieldRestrictions table holds fields that cannot be requested for this source by the Query Ticket message.

### Columns

Column	Type	Allow Nulls	Description
eGQueryTktFieldRestrictionID	Integer	N	Primary key, gateway counter
eGalaxySourceID	Integer	N	Foreign key to eGalaxySources table
FieldID	Integer	N	A numeric identifier for the lookup field assigned by eGalaxy <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PkeGQueryTktFieldRestrictionID	P	eGQueryTktFieldRestrictionID	Primary Key.

### <sup>1</sup> FieldID Values

Value	Gateway Constant Name	Description
1	VISUAL_ID_FIELD_ID	Scan code or barcode as read from scanner; combination of AccessCode and ID  (Ticket or Pass) (Required for <Query> otherwise optional)
2	ITEM_KIND_FIELD_ID	Specifies the kind of the item by looking at the Items.Kind using the PLU on the ticket record. (1- Regular/Ticket, 2- Pass)  (Ticket or Pass) (Optional)
3	STATUS_FIELD_ID	For a ticket, represents the current status of the ticket (0-Valid, 1-Voided, 2-Returned etc.)  For a pass, represents the current status of the pass (0- Valid, 1- Voided, 2- Returned, 3- Replaced, 4- Purchaser, 5- Expired)  (Ticket or Pass) (Optional)
4	EXCHANGEABLE_FIELD_ID	Set if the ticket is exchangeable  (Ticket Only) (Optional)
5	RETURNABLE_FIELD_ID	Set if the ticket is returnable  (Ticket or Pass) (Optional)
6	UPGRADEABLE_FIELD_ID	Set if the pass is upgradeable  (Pass only) (Optional)
7	RENEWABLE_FIELD_ID	Set if the pass is renewable  (Pass only) (Optional)
8	PRODUCT_NO_FIELD_ID	Product number from which the pass was sold  (Ticket or Pass) (Optional)
9	FKEY_NO_FIELD_ID	For ticket, combination of the ticket type's level and Fkey numbers within the product: FkeyNo = (Level * 100) + Fkey.  For pass, function key number from which the pass was sold  (Ticket or Pass) (Optional)
10	PLU_FIELD_ID	This is the PLU for the ticket or pass  (Ticket or Pass) (Optional)
11	PRICE_FIELD_ID	Price of the ticket or a pass  (Ticket or Pass)
12	REMAINING_PRICE_FIELD_ID	Remaining Price on the ticket  (Ticket only) (Optional)
13	TAX_FIELD_ID	Tax amount of the ticket or a pass  (Ticket or Pass) (Optional)
14	REMAINING_TAX_FIELD_ID	Remaining Tax amount on the ticket  (Ticket only) (Optional)
15	TAX_METHODS_FIELD_ID	The TaxMethods is an 8-character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket  (Ticket or Pass) (Optional)
16	COMPANY_FIELD_ID	Company this ticket or pass was sold from  (Ticket or Pass) (Optional)
17	TAX_FLAGS_FIELD_ID	An 8-character string containing an array of Y/N flags indicating which of the 8 possible taxes were applied when purchasing this ticket or pass  (Ticket or Pass) (Optional)
18	DISCOUNT_ID_FIELD_ID	Foreign key to Discounts.DiscountID, specifying the discount applied to this ticket, or 0 for no discount.  (Ticket or Pass) (Optional)
19	ACCESS_CODE_FIELD_ID	Access code of the ticket or a pass  (Ticket or Pass) (Optional)
20	ACCESS_CODE_NAME_FIELD_ID	Name of the AccessCode

21	TICKET_DATE_FIELD_ID	(Ticket or Pass) (Optional) Date when the ticket is valid for use (Date specific ticket)  (Ticket only) (Optional)
22	LOCKED_OUT_FIELD_ID	Set when the ticket or pass is locked out  (Ticket or Pass) (Optional)
23	LOCK_OUT_REASON_FIELD_ID	Reason if the ticket or pass is locked out (Lockouts.LockoutMsg)  (Ticket or Pass) (Optional)
24	USE_COUNT_FIELD_ID	Number of times the ticket or pass is used  (Ticket or Pass) (Optional)
25	REMAINING_USE_FIELD_ID	Remaining uses left on the ticket or a pass  (Ticket or Pass) (Optional)
26	UPDATE_STATUS_FIELD_ID	Update Status represented by the Ticket.UpdateCode  (Ticket only) (Optional)
27	NODE_NO_FIELD_ID	POS node number where the ticket or pass was sold  (Ticket only) (Optional)
28	TRANS_NO_FIELD_ID	The sequential ID number of the transaction (for a given NodeNo) from which the ticket belongs  (Ticket only) (Optional)
29	DATE SOLD_FIELD_ID	Date when ticket or pass was sold  (Ticket only) (Optional)
30	ORDER_ID_FIELD_ID	Order ID if ticket was sold in Order Entry  (Ticket only) (Optional)
31	CUSTOMER_ID_FIELD_ID	Customer ID of the Order for which ticket was sold  (Ticket only) (Optional)
32	CUST_NO_FIELD_ID	A string containing the account number of the customer that purchased this ticket. This column is blank if the ticket was not purchased by a customer with an account in the system, or if the ticket was added <i>dynamically</i> at the time of its first scan.  (Ticket only) (Optional)
33	EVENT_NO_FIELD_ID	Event ID of the event if ticket is an event ticket  (Ticket only) (Optional)
34	EVENT_NAME_FIELD_ID	Name of the event  (Ticket only) (Optional)
35	START_DATE_TIME_FIELD_ID	Date/time when event starts  (Ticket only) (Optional)
36	END_DATE_TIME_FIELD_ID	Date/time when event is over  (Ticket only) (Optional)
37	PASS_ID_FIELD_ID	Indicates the Galaxy pass ID value (Passes.PassNo)  (Pass only) (Optional)
38	PASS_ACCT_FIELD_ID	Pass account  (Pass only) (Optional)
39	FIRST_NAME_FIELD_ID	Pass holder's first name  (Pass only) (Optional)
40	MIDDLE_NAME_FIELD_ID	Pass holder's middle name/initial  (Pass only) (Optional)
41	LAST_NAME_FIELD_ID	Pass holder's last name  (Pass only) (Optional)
42	STREET_1_FIELD_ID	Street address line 1  (Pass only) (Optional)
43	STREET_2_FIELD_ID	Street address line 2  (Pass only) (Optional)
44	CITY_FIELD_ID	Pass holder's city  (Pass only) (Optional)
45	STATE_FIELD_ID	Pass holder's state  (Pass only) (Optional)
46	ZIP_FIELD_ID	Pass holder's ZIP/postal code  (Pass only) (Optional)
47	COUNTRY_CODE_FIELD_ID	Pass holder's country code  (Pass only) (Optional)

48	PHONE_FIELD_ID	Pass holder's telephone number (Pass only) (Optional)
49	EMAIL_FIELD_ID	Pass holder's e-mail address (Pass only) (Optional)
50	DOB_FIELD_ID	Pass holders' date of birth (Pass only) (Optional)
51	DATE_OPENED_FIELD_ID	Date when pass was opened (first issued) (Pass only) (Optional)
52	VALID_DAYS_FIELD_ID	Number of days the pass is valid (Pass only) (Optional)
53	KIND_FIELD_ID	PassKind ID (Pass only) (Optional)
54	VALID_UNTIL_FIELD_ID	The pass is valid until this day (Pass only) (Optional)
55	DATE_USED_FIELD_ID	Last date the pass was used (Pass only) (Optional)
56	MASTER_PASS_ID_FIELD_ID	ID of the primary pass of this pass (Pass only) (Optional)
57	PURCHASER_PASS_ID_FIELD_ID	ID of the purchaser pass of this pass (Pass only) (Optional)
58	MAX_PARTY_FIELD_ID	Max number of people that can be admitted with this pass (Pass only) (Optional)
59	REISSUE_COUNT_FIELD_ID	Number of times this pass was reissued (Pass only) (Optional)
60	USER_01_FIELD_ID	User definable field #1 (Pass only) (Optional)
61	USER_02_FIELD_ID	User definable field #2 (Pass only) (Optional)
62	USER_03_FIELD_ID	User definable field #3 (Pass only) (Optional)
63	USER_04_FIELD_ID	User definable field #4 (Pass only) (Optional)
64	USER_05_FIELD_ID	User definable field #5 (Pass only) (Optional)
65	USER_06_FIELD_ID	User definable field #6 (Pass only) (Optional)
66	USER_07_FIELD_ID	User definable field #7 (Pass only) (Optional)
67	USER_08_FIELD_ID	User definable field #8 (Pass only) (Optional)
68	USER_09_FIELD_ID	User definable field #9 (Pass only) (Optional)
69	USER_10_FIELD_ID	User definable field #10 (Pass only) (Optional)
70	LIMIT_COUNT_FIELD_ID	The number of times this pass was used to buy a pass-required ticket (Pass only) (Optional)
71	CATEGORY_FIELD_ID	Category number this pass belongs to (Pass only) (Optional)
72	SUB_CAT_FIELD_ID	Sub category number this pass belongs to (Pass only) (Optional)
73	VALUE_KIND_FIELD_ID	Meaning of the value in the "Value" column (0- Currency or Money, 1- Points) (Pass only) (Optional)
74	VALUE_FIELD_ID	Value on the pass (Pass only) (Optional)
75	FKEY_FLAGS_FIELD_ID	Function key options 1 to 8 (Y/N) (Pass only) (Optional)

76	FKEY_KIND_FIELD_ID	Function key kind (2-Pass, 6-Reissue, 7-Renewal, 13-Upgrade)  (Pass only) (Optional)
77	PRIOR_PASS_ACCT_FIELD_ID	Pass account number prior to upgrade/downgrade, renewal or reissue  (Pass only) (Optional)
78	PRIOR_PASS_KIND_FIELD_ID	Pass kind prior to upgrade/downgrade, renewal or reissue  (Pass only) (Optional)
79	PASS_NOTE_FIELD_ID	Any notes attached to the pass  (Pass only) (Optional)
80	MAXUSES_FIELD_ID	Maximum number of times the pass can be used  (Pass only) (Optional)

## 9.10 eGalaxySessions

The eGalaxySessions table stores active (and any recently expired) eGalaxy sessions.

### Columns

Column	Type	Allow Nulls	Description
eGalaxySessionID	Int	N	Primary key, always unique
ActiveIndicator	Int	Y	Indicates status of session's activity. <sup>1</sup>
LoginID	Int	Y	eGalaxy Login ID
SessionStart	DateTime	Y	Time of session start
SessionEnd	DateTime	Y	Time of session end
IPAddress	Varchar(40)	Y	IP address of session's client. Sized to accommodate IPv6 addresses (future)
HostName	Varchar(128)	Y	Host name from HTTP header
LastActivity	DateTime	Y	Indicates the date and time of last activity for this session

### Indexes

Name	Kind	Columns	Purpose
PkeGalaxySessionsGxSessionID	P	eGalaxySessionID	Primary key.
lxeGSessionsActiveIndicator		ActiveIndicator	Used by the query to get session records by their Status
lxeGSessionsLastActivity		LastActivity	Used by the query to check for idle sessions

<sup>1</sup> ActiveIndicator Values

Value	Gateway Constant Name	Description
0	N/A	Session is Inactive
1	N/A	Session is Active
2	SESSION_PENDING_IDLE	Session is Pending Idle

## 9.11 eGalaxySourceConfigOptions

This table holds configuration options for eGalaxy sources.

### Columns

Column	Type	Allow Nulls	Description
eGalaxySourceConfigOptionID	int	N	Unique Identifier
eGalaxySourceID	int	N	Unique ID of eGalaxy source this option refers to
Description	nvarchar(100)	N	Name for the configuration option
Value	nvarchar(1000)	N	Value for this configuration option, converted (if necessary) and stored as a string
DataType	int	N	Type of the configuration option (See possible values for ConfigurationOptions.DataType)
Code	int	N	Constant associated with a particular eGalaxy source configuration setting <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKeGalaxySourceConfigOptionID	P	eGalaxySourceConfigOptionID	Primary key

### <sup>1</sup> Code Values

Value	Gateway Constant Name	Description
0	EGALAXY_SOURCE_CONFIG_CODE_ENFORCE_AUTHORIZE_ELEMENT	Enforce <Authorize> element in all messages
1	EGALAXY_SOURCE_CONFIG_CODE_ENFORCE_GALAXY_PASSWORD_POLICY	Enforce Galaxy's password policies for new logons
2	EGALAXY_SOURCE_CONFIG_CODE_ENFORCE_GXUSER_LOCKOUTS	Enforce account lockouts for GxUser-based authentication
3	EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_ORDERFILEID	WebTemplate ID for order file email template
4	EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_ACTIVATIONFILEID	WebTemplate ID for activation file email template
5	EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_EODID	WebTemplate ID for EOD email template
6	EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_ERRORNOTIFICATIONID	WebTemplate ID for error notification email template
7	EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_PASSWORDREMINDERID	WebTemplate ID for password reminder email template
8	EGALAXY_SOURCE_CONFIG_CODE_PASSWORD_RESET_VALIDITY_PERIOD	Password Reset Validity Period (minutes)
9	EGALAXY_SOURCE_CONFIG_CODE_ALLOW_SPECIFYING_TICKET_VISUAL_IDS	Allow specifying the visual ID for tickets on the order line
10	EGALAXY_SOURCE_CONFIG_CODE_PREVENT_CANCELLATION_OF_INVOICED_ORDERS	Prevent cancellations of orders that have been invoiced
11	EGALAXY_SOURCE_CONFIG_CODE_USE_CORS_HEADERS	Indicates whether this source will return CORS compliant headers
12	EGALAXY_SOURCE_CONFIG_CODE_CORS_HEADER_ACCESS_CONTROL_ALLOW_ORIGIN	A comma separated list of allowed domains, or '*' if any domain is allowed to send a request    13   EGALAXY_SOURCE_CONFIG_CODE_ALLOW_QUERY_ORDERS_WITH_GROUP_VISIT_DATE_RANGE_ONLY   Allow QueryOrders with only the group visit date range    14   EGALAXY_SOURCE_CONFIG_CODE_BYPASS_ONLINE_EXCHANGE_RULES   Bypass the exchange rules configured in General Config or the PLU for the exchange window and exchange count limit    15   EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_ACTIVATION_CONFIRMATIONID   WebTemplate ID for activation confirmation email template    16   EGALAXY_SOURCE_CONFIG_CODE_TEMPLATE_ACTIVATION_CONFIRMATION_EMAIL_TEXT   Ticket Activation confirmation email text    17   EGALAXY_SOURCE_CONFIG_CODE_QUERYORDER_IDENTIFICATIONNO_VISIBILITY   IdentificationNo visibility for responses of the QueryOrder message    18   EGALAXY_SOURCE_CONFIG_CODE_QUERYCUSTOMERS_IDENTIFICATIONNO_VISIBILITY   IdentificationNo visibility for responses of the QueryCustomer message    19   EGALAXY_SOURCE_CONFIG_CODE_QUERYCONTACT_IDENTIFICATIONNO_VISIBILITY   IdentificationNo visibility for responses of the QueryContact message    20   EGALAXY_SOURCE_CONFIG_CODE_QUERYTICKET_IDENTIFICATIONNO_VISIBILITY   IdentificationNo visibility for responses of the QueryTicket message    21   EGALAXY_SOURCE_CONFIG_CODE_SHOW_LAST_N_IDENTIFICATIONNO   Show the last N characters of IdentificationNo where configured    23   EGALAXY_SOURCE_CONFIG_CODE_ALLOW_OVERRIDE_PRODUCT_DOWNGRADE_RULE   Allow the overriding of the Product Downgrade Rule    24   EGALAXY_SOURCE_CONFIG_CODE_FORCE_NEW_CONTACT_LASTNAME   Enforce last name for new contacts.    25   EGALAXY_SOURCE_CONFIG_CODE_APPLY_ORIGINALENDORSEMENT_ON_CANCEL   Apply the original endorsement on cancellation through the ActivateTicket MessageType    26   EGALAXY_SOURCE_CONFIG_CODE_ALLOW_UPDATING_ORDER_STATUS_FROM_CLOSE_TO_OPEN   Allow updating order status from closed to open.    29   EGALAXY_SOURCE_CONFIG_CODE_QUERYORDERS_IDENTIFICATIONNO_VISIBILITY   IdentificationNo visibility for responses of the QueryOrders message    30   EGALAXY_SOURCE_CONFIG_CODE_RESELLER_ENFORCE_STRICT_USER_AGENCY_VALIDATION   Enforce strict user/agency validation for reseller    31   EGALAXY_SOURCE_CONFIG_CODE_QUERYTICKET_ENABLE_SINGLE_PACKAGE_DETAIL_IN_RESPONSE   Enable single package detail in QueryTicket response option

## 9.12 eGalaxySourceIPAddresses

An eGalaxySourceIPAddresses table contains a list of IP addresses for an eGalaxy source. Column eGalaxySourceID is a foreign key to eGalaxySources table. The eGalaxy uses the list of IP addresses defined here to restrict messages from unknown clients.

Column	Type	Allow Nulls	Description
eGalaxySourceIPAddressID	Integer	N	Primary key. Obtained from GatewayCounters.
eGalaxySourceID	Integer	N	FK reference to eGalaxySources table
IPAddress	Varchar (128)	N	IP address of the computer (an eGalaxy source) that will be sending messages to eGalaxy

### Indexes

Name	Kind	Columns	Purpose
PKeGalaxySourceIPAddressID	P	eGalaxySourceIPAddressID	Primary Key.
IxeGSrcIPAdddeGalaxySrcID		eGalaxySourceID	Index, used to get IP addresses for a given eGalaxy source

### 9.13 eGalaxySourceMediaDefinitions

The eGalaxySourceMediaDefinitions table represents a detail list of media definitions associated with a specific eGalaxySource, used to recognize foreign media Visual ID's.

#### Columns

Column	Type	Allow Nulls	Description
eGalaxySourceMediaDefinitionID	Int	N	Primary key, always unique
eGalaxySourceID	Int	N	Foreign key to eGalaxySources.eGalaxySourceID
MediaID	Int	N	Foreign key to Media.MediaID

#### Indexes

Name	Kind	Columns	Purpose
PkeGSrcMediaDefID	P	EgalaxySourceMediaDefinitionID	Primary Key.
IxeGSrcMediaDefeGSourceID		EgalaxySourceID	Used for query to get all MediaID for the given eGalaxySourceID

## 9.14 eGalaxySources

The eGalaxySources table stores active configuration and control information relating to external sources of eGalaxyMessages.

### Columns

Column	Type	Allow Nulls	Description
eGalaxySourceID	Int	N	Primary key, always unique
ExternalSourceID	VarChar(128)	N	Matches SourceID field in messageheader. Must contain a unique value for each source.
SourceActive	Boolean	N	Messages from this source will be processed
OrderCustIDOverride	Int	Y	If non-zero, eGalaxy overrides the Customer ID element on each order from this source with the indicated value for Galaxy customer ID
OrderIDFromRange	Char(20)	Y	From range value for an <OrderID> to be valid. If the value is blank, <OrderID> is not validated for a range.
OrderIDThruRange	Char(20)	Y	Thru range value for an <OrderID> to be valid. If the value is blank, <OrderID> is not validated for a range.
EmailName	Text	Y	Email name of the client who will receive a summary status of the imported files.
EmailAddress	Text	Y	Email Address of the client who will receive a summary status of the imported files.
EmailTemplate	Text	Y	Template for order file confirmation e-mail
CustomerGroupID	Integer	Y	(Optional) Specifies a customer group to define allowable customer IDs for this source. FK to CustomerGroup table.
SecureOrder	Bit	Y	Determines if this source should mark new orders as secure-orders can only be modified by an authorized users.
PayOnIssuance	Bit	Y	If 1, Pay On Issuance orders are allowed. Payment is not journalised.  If 0, Pay On Issuance orders are not allowed. Payment is journalised.
SendEmailAtEOD	Bit	Y	If 1, e-mail is always sent at the end of the day  If 0, no e-mail is sent at the end of the day
SendEmailImmediately	Bit	Y	If 1, e-mail is sent immediately after processing an order (real-time orders) or a file (file import)  If 0, no e-mail is sent after processing an order or a file
PerformAVS	Bit	Y	Set to include verification for AVS (Address Verification Service) as part of the authorization request
PerformCVN	Bit	Y	Set to include verification for CVN (Card Verification Number) as part of the authorization request
AVSVerificationMethod	Integer	Y	Method of address verification, used to determine what data to verify for AVS <sup>1</sup>
ErrorNotificationEmailName	Text	Y	Email name of the client who will receive the error notification email
ErrorNotificationEmailAddress	Text	Y	Email Address of the client who will receive the error notification email
ErrorNotificationEmailTemplate	Text	Y	Template for the error notification e-mail
AllowTicketActivation	Bit	Y	Set when the source is enabled to sent the ActivateTicket message requests
AllowQueryTicket	Bit	Y	Set when the source is enabled to process QueryTicket message requests
AllowUpdatePass	Bit	Y	Set when the source is enabled to sent the UpdatePass message requests
RejectSecureRequests	Bit	Y	Used to determine if this source should reject secure real-time requests.
RejectNonSecureRequests	Bit	Y	Used to determine if this source should reject non-secure real-time requests.
AllowCustomers	Bit	Y	Set when the source is enabled to sent the Customers message requests
AllowQueryCustomer	Bit	Y	Set when the source is enabled to send the QueryCustomer message requests
AllowContacts	Bit	Y	Set when the source is enabled to send the Contacts message requests
AllowQueryContact	Bit	Y	Set when the source is enabled to send the QueryContact message requests
PreventOrderCommandCreation	Bit	Y	Set to prevent the creation of OrderCommand records for WebOrderProcessor when processing orders
AllowAutoReplenishRequests	Bit	Y	Set when the source is enabled to send the requests related to the auto replenishment of the ticket
AllowQueryUsage	Bit	Y	Flag to determine if user can request usage
AllowQueryLineage	Bit	Y	Flag to determine if user can request lineage
AllowQueryContactHistory	Bit	Y	Set when the source is enabled to send the QueryContactHistory message request
ActivationFileEmailTemplate	Text	Y	Template for activation file confirmation e-mail
EODEmailTemplate	Text	Y	Template for end of day e-mail
ExcludeUsageInQueryTicketResp	Bit	Y	If set to 1, no usage data is sent in the QueryTicketResponse message
PasswordReminderEmailTemplate	Text	Y	Template for password reminder e-mail
EnableRE	Bit	Y	Set to enable the Raiser's Edge support for all passes sold by this eGalaxy source
REConfigID	Int	Y	Specifies the Raiser's Edge configuration record to use for this source. Foreign key to REConfig.REConfigID
ConfirmationEmailOption	Int	Y	Determines which e-mail address should be used when generating order confirmation e-mail <sup>2</sup>  0 = Order Contact email address  1 = Ship to Contact email
AllowNoCustMerchForConfEmail	Bit	Y	Determines if a confirmation e-mail can be sent through a customer that is not associated to a merchant.
ReplenishPaymentProcessorURL	Varchar (256)	Y	URL of the Galaxy's payment processor for authorizing payment when performing the Manual Replenishment of a ticket in eGalaxy
ReplyToEmailAddress	Text	Y	Reply-to field on email will be set to this address. If it is not blank, this field will override the ReplyToEmailAddress setting from the eGalaxyConfig table.
AlwaysUseItemDesc	Bit	Y	Set to use description from item definition when populating value in the OrderLines.Description column
SalesChannelID	Int	Y	Sales Channel for the source  Sales Channel used by eGalaxy server to enforce Capacity Limits. Capacity Limits are configured by Sales Channel on the Event. Any sales done via eGalaxy server (third party import or Web store sales) is subjected to the capacity limits defined for this sales channel.
AllowEncryptionKeyImport	Bit	Y	Boolean flag that determines whether or not this eGalaxySource can accept the EncryptionKey real-time message.
AllowCheckGroupSalesLimits	Bit	Y	When set to 1, this eGalaxy source can process the CheckGroupSalesLimits message
EnforceUniqueReferenceValues	Bit	Y	Determines whether or not unique reference values are enforced
AllowResellerRequests	Bit	Y	When set to 1, this source is allowed to send Reseller Web store specific requests. For example, Users, Nodes, UpdateResellerData, and CreateTransaction requests
AllowDebitRequests	Bit	Y	When set to 1, this source is allowed to send Debit related requests. For example ImportDebitTransaction.

AllowContractRequests	Bit	Y	When set to 1, this source is allowed to send Contract related requests. For example, UpdateContractStatus
DeleteLinesWhenCancelingOrder	Bit	Y	When set to 1, eGalaxy server deletes the unissued order lines from an order when processing Cancel order requests from this source
UsePhotoStatusOnPass	Bit	Y	Controls whether or not the Photo and Pass statuses will be synchronized
ContractStatementDelivery	Int	Y	Defines how statements are delivered to a guest  0 = Include in body of email ( default)  1 = Attach as a PDF  2 = Do not send statements
AllowQueryOrderRequests	Bit	Y	Set when the source is enabled to send the QueryOrder message request
AllowQueryOrdersRequests	Bit	Y	Set when the source is enabled to send the QueryOrders message request
IgnoreEmptyPassFields	Bit	Y	This field is used to determine how blank XML nodes affect the updating of pass information during a renewal. If True, XML fields with no value will not update the pass. When False (the default), empty XML pass fields will cause the Pass field to be cleared in the database.
AllowLoyaltyAccountRequests	Bit	Y	Determines whether or not the eGalaxySource can process LoyaltyAccount message requests
AllowQueryLoyaltyAccountRequests	Bit	Y	Determines whether or not the eGalaxySource can process QueryLoyaltyAccount message requests
AllowLoyaltyRedemptionRequests	Bit	Y	Determines whether or not the source allows LoyaltyRedemption messages
AllowLinkLoyaltyAccountRequests	Bit	Y	Determines whether or not the source allows LinkLoyaltyAccount messages.
AuthorizationMode	Integer	Y	The authorization mode - see values below <sup>3</sup>
CombineDuplicateContacts	Bit	Y	Determines whether or not the eGalaxy Source utilizes logic to combine duplicate contacts when importing an order. Defaults to True.
EnableInvoicing	Bit	Y	If this option is set to 1 then ticket activation messages from this source will create a transaction that may be invoiced (provided that "Auto payment" is enabled for this source and the auto-payment FOP is a charge account). Otherwise, ticket activation messages from this source will not create a transaction that may be invoiced.
IgnoreEmptyTagsForContactFields	Bit	Y	When IgnoreEmptyTagsForContactFields is set, the empty tags in Contact XML data will be filtered out during the UpdateFields generation.
FilterEndorsementForSearch	Bit	Y	If the bit is set, eGalaxy will not populate the OrderPayments.Endorsement column. It will record a searchable version of the original Endorsement value in the SearchEndorsementValue column.
AllowUpdateTickets	Bit	Y	Set when the source is enabled to sent the UpdateTickets message requests.
EncoderPluginID	Integer	Y	Foreign key to Plugins.PluginID. A non-zero value indicates encode request with selected plugin.
EnforceRequiredUserFields	Bit	Y	Set when the source should reject orders that do not contain user fields that are marked as required.
EnablePriceSchedulesInPriceValidation	Bit	Y	When this option is disabled (default) the item price is validated only for Items with PriceMethod = BASE_PRICE_PRICE_METHOD
eGalaxySourceGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**Indexes**

Name	Kind	Columns	Purpose
PkeGalaxySourcesGxSourceID	P	eGalaxySourceID	Primary key.
AkeGalaxySourcesGxExtSourceID	A	ExternalSourceID	Alternate key

**<sup>1</sup> AVSVerificationMethod Values**

Value	Description
1	Verify the cardholder's billing address and postal code
2	Verify the cardholder's postal code

**<sup>2</sup> ConfirmationEmailOption Values**

Value	Gateway Constant Name	Description
0	ORDER_CONTACT_CONF_EMAIL_OPTION	Confirmation e-mail is sent to the e-mail address on the Order contact
1	SHIPTO_CONTACT_CONF_EMAIL_OPTION	Confirmation e-mail is sent to the e-mail address on the Ship-to contact

**<sup>3</sup> AuthorizationMode Values**

Value	Gateway Constant Name	Description
0	amNotConfigured	This setting has not been configured
1	amNever	Payments will never be authorized from the source
2	amConditional	Payments will be authorized if no authorization code is sent to eGalaxy Server, otherwise payments will not be authorized
3	amAlways	Payments will always be authorized from the source

## 9.15 eGalaxyTransactionIDs

The eGalaxyTransactionIDs table is used to keep track of journal transaction numbers for specific nodes. This was originally only used for eGalaxy but has been extended to be used for additional applications such as Contract Processor.

### Columns

Column	Type	Allow Nulls	Description
eGalaxyConfigID	Int	N	Primary key, always unique
NodeNo	Int	N	Node number, always unique
LastTranNo	Int	Y	Latest Transaction Number in use for the node

### Indexes

Name	Kind	Columns	Purpose
PKeGalaxyTransactionID	P	eGalaxyTransactionID	Primary Key
AKeGalaxyTransactionIDsNodeNo	A	NodeNo	Unique Key

## 9.16 EmailStatistics

An EmailStatistics table is added to store the statistics, which are primarily used when sending the emails. The table contains columns to identify which entity has created these statistics. Column GroupIDType identifies the contents of the GroupID column. Table EmailStatistics contains a header record, and the actual statistics can be defined in the EmailStatisticDetails table.

Currently, this table is only used by eGalaxy for its internal processing and therefore no GUI is provided to edit data in this table.

Column	Type	Allow Nulls	Description
EmailStatisticID	Integer	N	Primary key. Obtained from GatewayCounters
eGalaxyNodeNo	Integer	N	Node number of eGalaxy Server owning this record.
GroupIDType	Integer	N	Identifies the contents of the GroupID column. <sup>1</sup>
GroupID	Integer	N	A numeric code uniquely identifying the entity to whom these statistics belong to
DateOfStatistics	DateTime	N	Date and time when the statistics were collected
MessageQueueID	Integer	N	FK reference to MessageQueue table
EmailSent	Bit	N	Set when an e-mail using these statistics is sent

### Indexes

Name	Kind	Columns	Purpose
PKEmailStatisticID	P	EmailStatisticID	Primary Key.
IXEMailStatsGroupTypeGroupID		GroupIDType, GroupID	Index, used by query to retrieve status of an statistics by group type
IXEStNodeGrpIDDateOfStatEMSent		NodeNo, GroupIDType, DateOfStatistics, EmailSent	Improve query when selecting e-mails to send at end of day

<sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	GroupID Meaning
0	eGALAXY_SOURCE_EMAIL	Zero. Statistics for the e-mail to be sent to the eGalaxy source. The GroupID is the eGalaxySourceID.

## 9.17 EmailStatisticDetails

An EmailStatisticDetails table contains the actual statistics. Column EmailStatisticID is a foreign key to EmailStatistics table. Currently, eGalaxy uses the statistics stored in this table to send email to the eGalaxy source. This table will be used for internal processing and therefore no GUI is needed to edit data in this table.

Column	Type	Allow Nulls	Description
EmailStatisticDetailID	Integer	N	Primary key. Obtained from GatewayCounters
EmailStatisticID	Integer	N	FK reference to EmailStatistics table
Name	Varchar(256)	N	Name of the stat
Value	Varchar(256)	N	Value of the stat
HasCollection	Bit	Y	If 1, this statistic is a collection item (has more than one value) and the values for this statistic are in the EmailStatisticSubDetails table If 0, this statistic is NOT a collection item and value is in the Value column of this table
StatisticType <sup>1</sup>	Int	Y	Type of the statistic

### Indexes

Name	Kind	Columns	Purpose
PKEmailStatisticDetailID	P	EmailStatisticDetailID	Primary Key.
IXEmailStatDetailsEMailStatID		EmailStatisticID	Index, used by query to retrieve stats for given EmailStatisticID

### <sup>1</sup> StatisticType Values

Value	Gateway Constant Name	Description
0	ORDER_TRANS_TYPE	Statistic is for an order
1	ACTIVATION_TRANS_TYPE	Statistic is for activation

## 9.18 EmailStatisticSubDetails

This table is used to store details of the statistics that can have multiple values. Table contains a link to the EmailStatisticDetails table.

### Columns

Column	Type	Allow Nulls	Description
EmailStatisticSubDetailID	Int	N	Primary key. Obtained from GatewayCounters
EmailStatisticDetailID	Int	N	FK reference to EmailStatisticDetails table
Value	Varchar(256)	N	Value of the statistic

### Indexes

Name	Kind	Columns	Purpose
PKEmailStatSubDtlID	P	EmailStatisticSubDetailID	Primary Key.
IXEmailStatSubDtEmailStatDtlID		EmailStatisticDetailID	Index, used by the query to retrieve details for a given EmailStatisticDetailID

## 9.19 ErrorNotificationOptions

An ErrorNotificationOptions table contains the notification options for different eGalaxy error codes. Notification options defined here are used to notify user of the problem when a particular error code is encountered while importing orders in eGalaxy.

### Columns

Column	Type	Allow Nulls	Description
ErrorNotificationOptionID	Int	N	Primary key, always unique
eGalaxyNodeNo	Int	N	eGalaxy Node No the option belongs to
ErrorCode	Int	N	Error Code generated by eGalaxy
Action	Int	N	Action to take for ErrorCode <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKErrNotifOptErrNotifOptID	P	ErrorNotificationOptionID	Primary Key.
IXErrNotOptionseGalaxyNodeNo		eGalaxyNodeNo	Index to filter the error codes by eGalaxyNodeNo

### <sup>1</sup> Action Values

Value	Gateway Constant Name	Description
0	NOTIFY_ACT_eGALAXY_LOG	The error and the order information is added to the eGalaxy.log file
1	NOTIFY_ACT_ERROR_TABLE	Option to insert the error to eGalaxyErrorLog table
2	NOTIFY_ACT_ERROR_FILE	The error and the order information is added to the Error Description file. All the errors for this action are sent to the configured notification recipient at the End Of Day processing.
3	NOTIFY_ACT_EMAIL	The error and the order information is immediately sent within an e-mail using the Error Notification e-mail Template defined on the eGalaxy source

## 9.20 GxProcessTriggers

Stores records to allow WebOrderProcessor to add GxTrigger records for tickets and/or passes for specified processes.

### Columns

Column	Type	Allow Nulls	Description
GxProcessTriggerID	Int	N	Primary key, always unique
ProcessCode <sup>1</sup>	Int	Y	Process that this record applies to
NodeNumber	Int	Y	Associated node number
TriggerOnTicket	Bit	Y	If 1, a GxTrigger record will be created when a ticket is inserted
TriggerOnPass	Bit	Y	If 1, a GxTrigger record will be created when a pass is inserted
TriggerOnUsage	Bit	Y	If 1, a GxTrigger record will be created when a usage is inserted (when importing a usage record in ImportUsage eGalaxy server message)

### Indexes

Name	Kind	Columns	Purpose
PKGxProcessTriggerID	P	GxProcessTriggerID	Primary Key.

### <sup>1</sup> ProcessCode Values

Value	Gateway Constant Name	Description
1	PC_MAGICQUEST	Record is for the Magic Quest Process (Mserver)
2	PC_HANDSHAKE	Record is for the Handshake Process (HSServer)

## 9.21 HSConfig

Stores HSServer specific configuration settings.

### Columns

Column	Type	Allows Nulls	Description
HSConfigID	Int	N	Primary key, always unique
NodeNumber	Int	N	Node number, always unique
HandshakeVersion	Char(25)	N	Handshake version
TicketingSystemID	Char(50)	N	Ticketing System ID
ReceiverID	Char(50)	N	Receiver ID
HostAddress	Char(50)	N	Server name or IP address of the Handshake Server
Port	Int	N	Port to use when connection to the Handshake Server
PortTimeout	Int	N	Port Timeout (in milliseconds)
ScanInterval	Int	N	Number of seconds to wait between checks for new GxTrigger entries
ScanCount	Int	N	Number of record to process at one time
MaxLogSize	Int	N	Size in bytes that the log file can grow to before truncating log data. Set to 0 for unlimited
SendAlerts	Bit	N	If 1, Alerts will be sent to MWS when the Handshake server is down
MaxRecordsPerMessage	Int	Y	This indicates the maximum number of ticket and pass records that may be added to a single request message to Handshake by HSServer. A value of 0 causes the maximum number sent per message to be 1, which was the previous default.
RejectedTriggerInterval	Int	Y	The number of seconds to wait between checks for rejected GxTrigger records for automatic reprocessing.
SendImages	Bit	Y	Indicates if pass image data will be sent to Skidata's Handshake by HSServer.

### Indexes

Name	Kind	Columns	Purpose
PKHSConfigID	P	HSConfigID	Primary Key

## 9.22 OnlineExchangeHistory

The OnlineExchangeHistory table contains a record for each time a particular VisualID was exchanged online.

### Columns

Column	Type	Allows Nulls	Description
OnlineExchangeHistoryID	Int	N	Primary key, always unique
VisualID	nvarchar(40)	N	The ticket that was exchanged.
ExchangeDateTime	Char(25)	N	The date and time that the exchange occurred.
Action	Int	Y	The type of online exchange record. <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOnlineExchangeHistoryID	P	OnlineExchangeHistoryID	Primary Key
IXVisualIDExchangeDateTime		VisualID, ExchangeDateTime	Speed up querying all the exchanges that occurred for a VisualID.

### <sup>1</sup> Action Values

Value	Gateway Constant Name	Description
0		Exchange
1		Activation

### 9.23 RejectedOrderCommandErrors

Contains the errors for the ordercommands rejected by WebOrderProcessor. The table has ErrorCode and ErrorText columns, which contains the details of the error from WebOrderProcessor.

#### Columns

Column	Type	Allow Nulls	Description
RejectedOrderCommandErrorID	Integer	N	Primary key, gateway counter
OrderCommandID	Integer	N	FK reference to OrderCommands table
ErrorCode	Integer	N	Error code
ErrorText	Text	Y	Error text

#### Indexes

Name	Kind	Columns	Purpose
PKRejOrderCommandErrorsErrorID	P	RejectedOrderCommandErrorID	Primary Key, gateway counter
IXRejOrdCommErrsOrderCommandID		OrderCommandID	Used by the query to get errors for a given OrderCommandID

## 9.24 SequenceNumbers

It's used to store ticket / pass sequence numbers. The numbers are stored by node and company since each node can produce tickets for multiple companies. Currently, the table is only used by eGalaxy (when assigning a VisualID to a ticket) but will likely be used for POS machines in the future.

### Columns

Column	Type	Allow Nulls	Description
SequenceNumberID	Int	N	Primary key, always unique
Node	Int	Y	The node number for which this chain of sequence numbers is used for.
Company	Int	Y	Each node can use sequence numbers for multiple companies, each company sequence number starts at 1
SequenceNumber	Int	Y	The actual sequence number. The value is the LAST number used by a node for the specified company.

### Indexes

Name	Kind	Columns	Purpose
PKSequenceNumberID	P	SequenceNumberID	Primary Key.
IXSequenceNumbersNodeCompany	IX	Node, Company (includes SequenceNumber)	Used to query for a unique sequence number and speed up the creation of a new unique SequenceNumber value. Speed query performance.

**10 Event Store**

## 10.1 GxEventStore

Stores system events that can be used to update data and track history.

### Columns

Column	Type	Allow Nulls	Description
GxEventStoreID	bigint	N	Primary key, always unique, Identity.
SchemaVersion	nvarchar(250)	N	Version of the schema that the data attribute adheres to.
SchemaUrl	nvarchar(500)	Y	A link to the schema that the data attribute adheres to.
EventID	uniqueidentifier	N	Alternate primary key.
EventType	nvarchar(500)	N	Type of occurrence which has happened. <sup>1</sup>
EventTypeVersion	nvarchar(250)	Y	Version of the event which occurred.
EventTime	datetimeoffset(7)	N	UTC DateTimeOffset when the event occurred.
Source	nvarchar(500)	N	Producer of the event.
ContentType	nvarchar(250)	Y	Content type of the 'data' attribute value.
Data	nvarchar(max)	N	The event payload. It is encoded into a media format which is specified by the 'ContentType' attribute
Extensions	nvarchar(max)	Y	Extensions registered with this event. Usually additional metadata.

### Indexes

Name	Kind	Columns	Purpose
PKEventStore	P	GxEventStoreID	Primary key
IXEventIDEventTypeEventTime	A	EventID,EventType,EventTime	
IXEventType	A	EventType	

### <sup>1</sup> EventType Values

Value	Description
AcReservationInventoryUpdated	Generated any time an admission reservation is created or cancelled.
AcReservationNotificationCreated	Generated any time an email notification occurs.
AcReservationSummaryCreated	Generated at the end of each day to supply a summary count of reservation changes.
AdmissionReservationCancelled	Generated any time an admission reservation is cancelled on behalf of the reservation holder. The event contains the state of the reservation before and after the cancellation occurred.
AdmissionReservationCreated	Generated each time a new admission reservation is created.
AdmissionReservationForgaveCancellation	Generated any time an admission reservation cancellation is forgiven. The event contains the state of the reservation before and after the cancellation forgiveness occurred.
AdmissionReservationForgaveNoShow	Generated any time an admission reservation no show is forgiven. The event contains the state of the reservation before and after the no show forgiveness occurred.
AdmissionReservationModified	Generated any time an admission reservation is modified. The event contains the state of the reservation before and after the modification occurred.
AdmissionReservationMoved	Generated any time an admission reservation is moved from one pass holder to another. The event contains the state of the reservation before and after the reservation was moved.
AdmissionReservationNoShowed	Generated any time an admission reservation is marked as a no show. This occurs when a reservation was booked for a particular date but the reservation was never used for admission. The event contains the state of the reservation before and after the no show occurred.
AdmissionReservationReactivated	Generated any time an admission reservation is set back to the active status. The event contains the state of the reservation before and after the reactivation occurred.
AdmissionReservationSystemCancelled	Generated any time an admission reservation is cancelled by the system. The event contains the state of the reservation before and after the cancellation occurred.
AdmissionReservationUsageReapplied	Generated any time the usage for an admission reservation is reapplied. The event contains the state of the reservation before and after the usage was reapplied.
AdmissionReservationUsed	Generated any time an admission reservation is marked as used. This occurs. The event contains the state of the reservation before and after the cancellation forgiveness occurred.
AdmissionReservationProfileBlocked	Generated each time an admission reservation profile is blocked. The event contains the state of the profile before and after the block occurred.
AdmissionReservationProfileCreated	Generated each time a new admission reservation profile is created.
AdmissionReservationProfileUnlocked	Generated each time an admission reservation profile is unlocked. The event contains the state of the profile before and after the unlock occurred.
AdmissionReservationProfileConsequenceCreated	Generated each time a new admission reservation profile consequence is created.
AdmissionReservationProfileConsequenceEnforced	Generated each time an admission reservation profile consequence is enforced. The event contains the state of the consequence before and after the consequence was enforced.
AdmissionReservationProfileConsequenceExpired	Generated each time an admission reservation profile consequence is expired. The event contains the state of the consequence before and after the consequence was expired.
AdmissionReservationProfileConsequenceInfractionCreated	Generated each time a new admission reservation profile consequence infraction is created.
AdmissionReservationProfileConsequenceInfractionExpired	Generated each time an admission reservation profile consequence infraction is expired. The event contains the state of the infraction before and after the expiration occurred.
AdmissionReservationProfileConsequenceInfractionForgiven	Generated each time an admission reservation profile consequence infraction is forgiven. The event contains the state of the infraction before and after the forgiveness occurred.
AdmissionReservationProfileConsequenceInfractionRemoved	Generated each time an admission reservation profile consequence infraction is removed. The event contains the state of the infraction before and after the removal occurred.
PassCreated	Generated each time a pass is created.
PassModified	Generated each time a pass is modified. The event contains the state of the pass before and after the modification occurred.
PassReissued	Generated each time a pass is reissued. The event contains the state of the pass before and after the reissue occurred.
PassRenewed	Generated each time a pass is renewed. The event contains the state of the pass before and after the renewal occurred.
PassReprinted	Generated each time a pass is reprinted. The event contains the state of the pass before and after the reprint occurred.

PassReturned	Generated each time a pass is returned. The event contains the state of the pass before and after the return occurred.
PassUpgraded	Generated each time a pass is upgraded. The event contains the state of the pass before and after the upgrad occurred.
PassVoided	Generated each time a pass is voided. The event contains the state of the pass before and after the void occurred.
TicketToPassUpgrade	Generated each time a ticket is upgraded to a pass. The event contains the ticket that was upgraded and the pass that it was upgraded to.
UsageCreated	Generated each time usage is created.

## 10.2 GxEventStoreConsumerDetails

Configuration details for each consumer which tells the consumer which events to read from the event store.

### Columns

Column	Type	Allow Nulls	Description
GxEventStoreConsumerDetailID	int	N	Primary key, always unique, Identity
EventText	nvarchar(500)	N	Text filter used to determine which events a consumer should read
ConsumerDetailKind	int	N	Type of text filter <sup>1</sup>
GxEventStoreConsumerGUID	uniqueidentifier	N	Foreign key to GxEventStoreConsumer
GxEventStoreConsumerDetailGUID	uniqueidentifier	N	Alternate primary key

### Indexes

Name	Kind	Columns	Purpose
PKGxEventStoreConsumerDetails	P	GxEventStoreConsumerDetailID	Primary key

<sup>1</sup> ConsumerDetailKind Values

Value	Description
1	StartsWith match - Matches on the namespace of the event using "starts with" logic.
2	Event Name match - Matches on the exact event name.

### 10.3 GxEventStoreConsumers

The list of processes that are configured to consume events from the event store.

#### Columns

Column	Type	Allow Nulls	Description
GxEventStoreConsumerID	int	N	Primary key, always unique, Identity
ConsumerName	nvarchar(255)	N	Name of process that is consuming events from the event store table
GxEventStoreConsumerGUID	uniqueidentifier	N	Alternate primary key

#### Indexes

Name	Kind	Columns	Purpose
PKGxEventStoreConsumers	P	GxEventStoreConsumerID	Primary key

## 10.4 GxEventStoreQueue

Contains information about GxEventStore records for processing by external services.

### Columns

Column	Type	Allow Nulls	Description
GxEventStoreQueueID	int	N	Primary key, always unique, Identity.
EventID	uniqueidentifier	N	Foreign Key to GxEventStore
ConsumerName	nvarchar(100)	N	Name of the process that wrote and consumed this item from the queue
EventType	nvarchar(500)	N	Name of the event. Taken from EventType in GxEventStore
NodeID	int	Y	Node number that created this record
ProcessStatus	int	N	Current status of this record <sup>1</sup>
CreatedAt	datetimeoffset(7)	N	Date and time this record was created
FinishedAt	datetimeoffset(7)	Y	Date and time this record finished processing
Tries	int	N	Number of attempts that have been made to process this record
LastError	nvarchar(max)	Y	Last known error if the record fails to process
GxEventStoreQueueGUID	uniqueidentifier	N	Alternate primary key

### Indexes

Name	Kind	Columns	Purpose
PKGxEventStoreQueueID	P	GxEventStoreQueueID	Primary key
IXEventIDConsumerNameEventType	A	EventID,ConsumerName,EventType	
IXConsumerName	A	ConsumerName	
IXProcessStatusConsumerName	A	ProcessStatus,ConsumerName	

<sup>1</sup> ProcessStatus Values

Value	Description
0	Available - Event has been written to the processing queue and is waiting to process.
1	Processing - Event is currently being processed.
2	Error - Event has been processed but resulted in an error.
3	Complete - Event has been successfully processed.

## 10.5 GxEventStoreTags

Stores keywords or IDs related to a specific event.

### Columns

Column	Type	Allow Nulls	Description
GxEventStoreTagID	int	N	Primary key, always unique, Identity.
EventID	uniqueidentifier	N	Foreign Key to GxEventStore
Tag	nvarchar(255)	N	Text value tag associated with this event
GxEventStoreTagGUID	N		Alternate primary key

### Indexes

Name	Kind	Columns	Purpose
PKGxEventStoreTagID	P	GxEventStoreTagID	Primary key
IXEventID	A	EventID	
IXTag	A	Tag	

## 11 Gift

## 11.1 Appeals

This table holds information regarding appeals for Gifts.

Columns

Column	Type	Allow Nulls	Description
AppealID	Int	N	Primary key, always unique
Description	Varchar(100)	N	Description of the appeal
Inactive	Bit	Y	Indicates if the appeal is currently inactive
DiscountID	Int	Y	References Discounts SQL table
AppealCode	Varchar(20)	Y	A unique alphanumeric code for this appeal.
AppealGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

Indexes

Name	Kind	Columns	Purpose
PKAppealID	P	AppealID	Primary Key
IXAppealsInactive		Inactive	Speeds up loads of inactive appeals

## 11.2 AppealSolicitationConnections

This table contains connections between Appeals and Solicitations table

Columns

Column	Type	Allow Nulls	Description
AppealSolicitationConnectionID	Int	N	Primary key, always unique
AppealID	Int	N	Foreign key to Appeals.AppealID
SolicitationID	Int	N	Foreign key to Solicitations.SolicitationID

Indexes

Name	Kind	Columns	Purpose
PKAppealSolicitationConnectionID	P	AppealCampaignConnectionID	Primary Key.
IXAppealID		AppealID	Load connections via AppealID
IXSolicitationID		SolicitationID	Load connections via SolicitationID

### 11.3 Campaigns

This table holds information regarding the Campaigns used for Gifts.

#### Columns

Column	Type	Allow Nulls	Description
CampaignID	Int	N	Primary key, always unique. Value obtained from GatewayCounters table
Description	Varchar(100)	N	Description of the campaign
Inactive	Bit	Y	Indicates if the campaign is currently inactive

#### Indexes

Name	Kind	Columns	Purpose
PKCampaignID	P	CampaignID	Primary Key
IXCampaignsInactive		Inactive	Speeds up loads of inactive campaigns

## 11.4 CampaignAppealConnections

This table contains connections between Campaigns and Appeals tables.

Columns

Column	Type	Allow Nulls	Description
CampaignAppealConnectionID	Int	N	Primary key, always unique
CampaignID	Int	N	Foreign key to Campaigns.CampaignID
AppealID	Int	N	Foreign key to Appeals.AppealID

Indexes

Name	Kind	Columns	Purpose
PKCampaignAppealConnectionID	P	CampaignAppealConnectionID	Primary Key.
IXAppealID		AppealID	Load connections via AppealID
IXCampaignID		CampaignID	Load connections via CampaignID

## 11.5 Funds

This table holds information regarding the different kind of funding for Gifts.

### Columns

Column	Type	Allow Nulls	Description
FundID	Int	N	Primary key, always unique
Description	Varchar(100)	N	Description of the fund
Inactive	Bit	Y	Indicates if the fund is currently inactive

### Indexes

Name	Kind	Columns	Purpose
PKFundID	P	FundID	Primary Key
IXFundsInactive		Inactive	Speeds up loads of inactive funds

## 11.6 GiftAidCategoryPriceData

The GiftAidCategoryPriceData table is used by the eGalaxy web store to quickly access pricing by category for gift aid items for web page display purposes. This table is generated and sent during a publish to the eGalaxy web store.

### Columns

Column	Type	Allow Nulls	Description
GiftAidCategoryPriceDataID	Int	N	Primary key, always unique
CategoryID	Int	N	FK to SalesChannelDetails.SalesChannelDetailID when SalesChannelDetails.DetailType = 1(ID_SALES_CAT_DETAIL), the Category that this entry relates to.
PLU	Char(20)	N	FK to Items.PLU, the ticket/item that this entry relates to.
Price	Money	Y	The price for 1 ticket/item as calculated by Galaxy.
Discount	Money	Y	The discount amount for 1 ticket/item as calculated by Galaxy.
Tax	Money	Y	The tax amount for 1 ticket/item as calculated by Galaxy.
GiftAidCategoryPriceDataGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKGiftAidCategoryPriceDataID	P	GiftAidCategoryPriceDataID	Primary Key

## 11.7 GiftAidClaims

This table holds information regarding GiftAid transactions processed by Galaxy and whether they have been claimed by the vendor yet.

Columns

Column	Type	Allow Nulls	Description
GiftAidClaimID	Int	N	Primary key, always unique. Value obtained from GatewayCounters table.
JnlTranID	Int	N	FK to the JnlHeaders.JnlTranID.
Claimed	Bit	N	Inserted as 0 (False) by Galaxy for qualifying transactions. Vendors may set this to 1 (True) after processing the claim.
ClaimDateTime	DateTime	Y	The time that the claim was processed by the vendor.
UserID	Int	Y	FK to GxUsers.GxUserID.
ClaimName	NVarChar(max)	Y	Vendor-defined data for use in the GiftAid claims process.

Indexes

Name	Kind	Columns	Purpose
PKGiftAidClaimID	P	GiftAidClaimID	Primary Key
IXGiftAidClaimsJnlTranID	A	JnlTranID	
IXGiftAidClaimsClaimed	A	Claimed	
IXGiftAidClaimsClaimDateTime	A	ClaimDateTime	

## 11.8 Gifts

This table holds information regarding Gifts.

Columns

Column	Type	Allow Nulls	Description
GiftID	Int	N	Primary key, always unique. Value obtained from GatewayCounters table
TransactionNumber	Int	N	The transaction number associated with this gift
NodeID	Int	N	The Node Number where the transaction was completed
ContactID	Int	N	References CustContacts.CustContactID
GiftDate	DateTime	Y	Date of the transaction in which the gift was sold
REGiftType	Int	Y	Gift type from the Raiser's Edge gift record. Uses the value from the TYPE field in the Raiser's Edge Gift table. <sup>1</sup>
OwnerId	Int	Y	The GiftID of the Gift's owner. For Pledge payment gifts, the OwnerID indicates the Pledge gift that the payment should be applied to.

Indexes

Name	Kind	Columns	Purpose
PKGiftID	P	GiftID	Primary Key.
IXGiftsContactID to Gifts table	A	ContactID	Index used for loading gifts by contact ID

<sup>1</sup> REGiftType Values

Value	Gateway Constant Name	Description
1	RE_GIFT_TYPE_CASH	Cash
2	RE_GIFT_TYPE_PAY_CASH	Pay-Cash
3	RE_GIFT_TYPE_MG_PAY_CASH	MG Pay-Cash
8	RE_GIFT_TYPE_PLEDGE	Pledge
9	RE_GIFT_TYPE_STOCK	Stock/Property
10	RE_GIFT_TYPE_STOCK SOLD	Stock/Property (Sold)
11	RE_GIFT_TYPE_PAY_STOCK	Pay-Stock/Property
12	RE_GIFT_TYPE_MG_PAY_STOCK	MG Pay-Stock/Property
13	RE_GIFT_TYPE_PAY_STOCK SOLD	Pay-Stock/Property (Sold)
14	RE_GIFT_TYPE_MG_PAY_STOCK SOLD	MG Pay-Stock/Property (Sold)
15	RE_GIFT_TYPE_GIFT_IN_KIND	Gift-in-Kind
16	RE_GIFT_TYPE_PAY_GIFT_IN_KIND	Pay-Gift-in-Kind
17	RE_GIFT_TYPE_MG_PAY_GIFT_IN_KIND	MG Pay-Gift-in-Kind
18	RE_GIFT_TYPE_OTHER	Other
19	RE_GIFT_TYPE_PAY_OTHER	Pay-Other
20	RE_GIFT_TYPE_MG_PAY_OTHER	MG Pay-Other
21	RE_GIFT_TYPE_WRITE_OFF	Write Off
22	RE_GIFT_TYPE_MG_WRITE_OFF	MG Write Off
27	RE_GIFT_TYPE_MG_PLEDGE	MG Pledge
28	RE_GIFT_TYPE_ADJUSTMENT	Adjustment
30	RE_GIFT_TYPE_RECURRING_GIFT	Recurring Gift
31	RE_GIFT_TYPE_RECURRING_GIFT_PAY_CASH	Recurring Gift Pay-Cash
32	RE_GIFT_TYPE_GL_REVERSAL	GL Reversal
33	RE_GIFT_TYPE_AMENDMENT	Amendment
34	RE_GIFT_TYPE_PLANNED_GIFT	Planned Gift
1	RE_GIFT_TYPE_CASH	Cash

## 11.9 GiftDetails

This table holds information regarding the details of Gifts.

Columns

Column	Type	Allow Nulls	Description
GiftDetailID	Int	N	Primary key, always unique. Value obtained from GatewayCounters table
GiftID	Int	N	Reference to the Gifts table. There can be many records in this table for the same GiftID
PLU	VarChar(20)	Y	The PLU of the item that the gift was recorded for.
VisualID	VarChar(40)	Y	VisualID of ticket if the gift is associated to a ticket with a VisualID(Barcode)
Amount	Money	Y	Amount of this gift. This will be the non-taxed amount.
Status	Int	Y	Indicates the status of the gift detail. <sup>1</sup>
CampaignID	Int	Y	Foreign key link to the Campaigns table. Indicates the campaign recorded for gifts.
FundID	Int	Y	Foreign key link to the Funds table. Indicates the fund recorded for gifts.
AppealID	Int	Y	Foreign key link to the Appeals table. Indicates the appeal recorded for gifts.
SolicitationID	Int	Y	Foreign key link to the Solicitations table. Indicates the solicitation recorded for gifts.
OrderLineID	Int	Y	The unique ID of the order line the gift detail was sold with.
ContractID	Int	Y	Link to PaymentContracts table

Indexes

Name	Kind	Columns	Purpose
PKGiftDetailID	P	GiftDetailID	Primary Key.
IXGiftDetailsCampaignID		CampaignID	Used for loading gift details by CampaignID
IXGiftDetailsAppealID		AppealID	Used for loading gift details by AppealID
IXGiftDetailsFundID		FundID	Used for loading gift details by FundID
IXGiftDetailsSolicitationID		SolicitationID	Used for loading gift details by SolicitationID
IXGiftDetailsVisualID		VisualID	Used for loading gift details by VisualID.
IXGiftDetailsOrderLineID		OrderLineID	Used for loading gift details by OrderLineID

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	VALID_GIFT	Valid gift record
1	VOIDED_GIFT	Voided gift record
2	RETURNED_GIFT	Returned gift record

## 11.10 GiftFieldsResponses

This table holds responses to the User Defined Fields of Gifts.

Columns

Column	Type	Allow Nulls	Description
GiftFieldsResponseID	Int	N	Primary key, always unique. Value obtained from GatewayCounters table
GiftID	Int	N	Reference to the Gifts table. There can be many records in this table for the same GiftID
UserFieldID	Int	N	Reference to the GxUserFields table. Every record will have a unique combination of GiftID-UserFieldID
Response	VarChar(40)	Y	The Response string for the specific User Field

Indexes

Name	Kind	Columns	Purpose
PKGiftFieldsResponseID	P	GiftFieldsResponseID	Primary Key.

### 11.11 Solicitations

This table holds information regarding solicitations/packages for Gifts.

Columns

Column	Type	Allow Nulls	Description
SolicitationID	Int	N	Primay key, always unique. Value obtained from GatewayCounters table
Description	Varchar(100)	N	Description of the solicitation
Inactive	Bit	Y	Indicates if the solicitation is currently inactive

Indexes

Name	Kind	Columns	Purpose
PKSolicitationID	P	SolicitationID	Primary Key
IXSolicitationsInactive		Inactive	Speeds up loads of inactive solicitations

## 12 Journal

The basic scheme of translating AX journal records to a Server database consists of :

- Writing a Journal Header entry in the JnlHeaders table for each transaction
- Culling off the financial data from every type of AX journal record and entering it into the JnlDetails table
- Writing any leftover information from the AX journal record into auxiliary tables when necessary

### Journal Header Exceptions

Since they do not contain detail information, but are necessary to trace, Poll records are treated as if they were journal headers and an entry is made into the JnlHeaders table. The NodeNo represents the node that was polled, TransNo is always zero, TransDate is always the date of the poll, and the PollID is placed in the Reference column. No entries are made in any other tables for Poll records.

Only valid transactions are written to Server. Transactions that do not balance are not written to the Server but an exception notice is written to the DBSync.LOG file for later review so the user knows which transactions could not be written.

### Other Exceptions

#### Credit Memos

REFUND records journalized for credit memos (or Order Returns in AX) do not have auxiliary table information. The previous "JnlRefunds" table is no longer used by the system. Instead, credit memo information is available from the Order tables.

#### Deposits

Deposit records in AX have a free form text field referred to as the deposit id, which is padded with leading zeroes. Since this is a text field, these leading zeroes are stripped before being written to the JnlDeposits table. When we remove the unexpected leading zeroes, we remove them even if the user typed them into the field.

#### Events

EVENT records in AX are not written.

#### JnlDetails

OVER\_SHORT records do not have auxiliary table information, the amount is contained on the JnlDetails.

MEMO records only have Auxiliary table information, no financial info.

CAPACITY and EVENT records do not have auxiliary table information.

#### PaidIOs

PAIDOUT records have the signs (+/-) of their amounts flipped before they are placed into the JnlDetails.Amount column. On AX, the sign did not matter, but it is necessary to have the correct sign to get transactions to balance within the new environment.

#### Tickets

Ticket records in AX that are already found in the Server's Tickets table are processed as follows: The existing Tickets row is read and if the Status column is different from the AX record, and the AX record has a more current LastUse value, the existing Tickets row is updated for only the Status and LastUse columns.

#### Voids

Voided transactions on AX contain an exact copy of the original transaction, including any memo records. Since the memo records are superfluous, since the transaction is voided, the memo records of voided transactions are not written a second time to the Server.

## 12.1 JnlHeaders

The **JnlHeaders** table contains the header information (start of a transaction) for each journal transaction.

### Columns

Column	Type	Allow Nulls	Description
JnlTranID	Integer	N	Primary key, always unique.
NodeNo	Integer	N	Node number of the POS generating the transaction.
TranNo	Integer	N	Transaction number journalized from the POS.
FiscalDate	Datetime	N	Fiscal date this transaction applies to (there is no time component to the value).
TranDate	Datetime	N	date the transaction was generated on the source machine.
SeqNo	Integer	N	For internal use only - an arbitrary sequence number generated by Gateway to distinguish between transaction headers with the same TranDate.
JnlCode	Integer	N	Type of header record, see JnlCodeID Values under the JnlDetails table for more information.
Userld	Integer	N	User ID that made this transaction.
Agency	Integer	N	From the system configuration of the POS that made this transaction, this is the value of "Agency ID" multiplied by 1000, added with the value for "Window".
TranKind	Integer	N	Type of transaction, see table: <sup>1</sup>
Adjustment	Bit	N	System adjustment transaction.
ShiftNo	Integer	N	Shift number.
Reference	Char(10)	N	Poll ID (if JnlCode = 8), customer account number, or blank.
Posted	datetime	N	Date the transaction was posted to Server tables.
AdjustmentUser	Int	Y	ID of the user making a system adjustment. If this is not an adjustment this value will be zero
AdjustmentTime	DateTime	Y	Time set on back date or system adjustment interface. If this is not a back date or system adjustment transaction this value will be zero.
CompanyID	Int	Y	Company associated with the transaction
SupervisorID	Int	Y	ID of user who gave supervisor approval
SAReason	Int	Y	Read code for why supervisor approval was needed <sup>2</sup>
PostedToDB	Bit	Y	Indicates if the complete transaction has been posted. When Journal Sender reaches the end of a transaction and sends, this value is set to True.
ReceiptNo	Int	Y	Number unique per node that is incremented for each financial transaction (sales, returns, voids)
LoyaltyBonusPoints	Float	Y	Stores the bonus points awarded to the transaction.
TransUpsellCancelled	Bit	Y	Indicates if transactional upsell options were offered and all were rejected in the transaction.
TranslationLanguageID	Int	Y	Records the TranslationLanguageID under which the transaction was created. Foreign key to TranslationLanguages.TranslationLanguageID.
TransStartTime	DateTime	Y	Date and time when the transaction was started. For a sale transaction, this is the time when the first item was added to the transaction.
TransEndTime	DateTime	Y	Date and time when the transaction ended. This is the time when the EOT was recorded for the transaction. This is recorded in the journal on the EOT record.
TransSalesTotal	Money	Y	Total Amount of the transaction (Tickets, Items + Taxes)

### Indexes

Name	Kind	Columns	Purpose
PKJnlHeadersJnlTranID	P	JnlTranID	Primary key (non-clustered).
CXJnlHeadersTranDate		TranDate	Clustered index with 80% Fill Factor
AKJnlHeadersNodeTranDateJnlCde	A	NodeNo, TranNo, TranDate, JnlCode	Includes JnlTranID as a non-key field Alternate key. This key ensures no Galaxy transaction is duplicated
IXJnlHeadersNodeNoTranNoSeqNo		NodeNo, TranNo, SeqNo	Unknown.
IXJnlHeadersDateNodeTranSeq		TranDate, NodeNo, TranNo, SeqNo	Unknown.
IXJnlHeadersTranDateJnlCodeUserID		TranDate, JnlCodeID, UserID	For [Select4Browse] section of GTSJnlHeaders.SQL
IXJnlHdrsFiscAgcyTrnUsrNdShft		FiscalDate, Agency, TranNo, UserID, NodeNo, ShiftNo	Index to speed up SQL Agency and Shift reports from Galaxy.
IXJnlHdrResellerSalesReports		JnlCode, FiscalDate, Agency, UserID, NodeNo	Optional index for Reseller reports.
IXJnlHdrsTranUserFiscNodeTran		JnlTranID, UserID, FiscalDate, TranDate, NodeNo, TranNo	Optional index to improve database performance; to add this index run the GalaxyDatabase script with @AddIndexesForJNLTables set to 1
IXJnlHdrsFiscalDateTranKind		FiscalDate, TranKind	Only created when Smart Upsell is enabled to help with calculation of sales goals.
IXSQLShiftReport1	IX	FiscalDate, NodeNo, ShiftNo, UserID, Agency, JnlTranID, TranDate, TranNo, JnlCode, Adjustment, AdjustmentTime	Speed up SQL shift reports from Manager's Workstation

<sup>1</sup> **TranKind** Values

Value	Gateway Constant Name	Description
1	TICKET_TRANS	Ordinary sales or return transaction
3	TENDEX_TRANS	Tender exchange (TENDEX) transactions
4	VOID_TRANS	Void transactions
8	MODE_TRANS	Based on the value of JnlCode, records entry into Ticketing, Maintenance or Accounting mode
9	STARTUP_TRANS	Records activation of the ticketing system
10	DEPOSIT_TRANS	Deposits
11	OTHER_TRANS	Used for miscellaneous transactions, including logons, logoffs, change funds, cashouts, the "Print Ticket Set" function, moving event capacities, and activating events.
12	STOCK_TRANS	Records ticket stock usage before cashout (unless the "Suppress Stock Usage Calculation" option is selected in the Online Configuration)
13	USAGE_TRANS	Records usage records (normally Access Control scans)
14	BATCH_RETURN_NON_REV	Used for returning tickets issued for bulk exchange
15	RECEIPT_TRANS	Used for recording receipt function usage, if "Journalize Detail/Summary Receipts" options are selected
16	ORDER_PICKUP	Order Pickups

17	ORDER_UNPICKUP	Order Unpickups
18	SV_LOOKUP	Stored Value Lookup
19	VEHICLE_ENTRY_MEMO_TRANS	DVR: Vehicle triggered an Entrance
20	VEHICLE_EXIT_MEMO_TRANS	DVR: Vehicle triggered an Exit
21	DM_MEMO_TRANS	User entered Data Maint
22	BAD_LOGON_TRANS	A failed login attempt
23	VIEW_LOG_TRANS	Viewed Log
24	CCF_TRANS	CCF transaction
25	SETTLEMENT_TRANS	
26	TOGGLE_SURVEYS_TRANS	Transaction survey prompting was turned on or off by Galaxy function or MWS.
27	SUPERVISOR_APPROVAL_TRANS	Supervisor approval transaction
28	TOGGLE_VEHICLE_DETECTOR_BYPASS_TRANS	Toggle vehicle detector loop bypass on/off transaction
29	TOGGLE_VEHICLE_DETECTOR_ENABLE_TRANS	
30	REINITIALIZE_VEHICLE_DETECTOR_TRANS	
31	FISCAL_SUMMARY_TRANS	Records periodic summary of financial totals for a node.
32	TRANSACTION_CANCELED_TRANS	Records when a user cancels a transaction where products have already been selected.
33	ITEM_QUANTITY_REDUCED_TRANS	Records when a user reduces the quantity of a product in a transaction.
34	FISCAL_ARCHIVING_TRANS	Records audit details when a fiscal archive is generated.
35	REFUND_IN_DOWNGRADE_TRANS	Refund approved by supervisor in a downgrade situation where not refunding money was configured.
36	JOURNAL_LOG_TRANS	Used for logging information to the journal

**2 SAReson Values**

Value	Gateway Constant Name	Description
0	SA_UNDEFINED	Initial Value. No reason defined
1	SA_VOID	Transaction Void
2	SA_DEPOSIT	Deposit
3	SA_CASHOUT	Cashout
4	SA_ALTERNATE_RETURN_FOP	Select alternate return FOP
5	SA_TOGGLE_SURVEYS	Toggle survey prompting
6	SA_JOURNAL	View journal
7	SA_LEAD_MENU	Enter lead menu
8	SA_RETURN	Return
9	SA_COMP_TICKETS	Sell complimentary tickets
10	SA_RESTRICTED_TICKETS	Sell restricted tickets
11	SA_SALE	Sell tickets/items
12	SA_CREDIT_PROTOCOL_HANDSHAKE	Credit protocol handshake
13	SA_X_RECEIPT	Print X Receipt
14	SA_DUP_REFERENCE_TICKET	Duplicate reference ticket
15	SA_VIEW_EXP_CASHOUT_TOTALS	View expected cashout totals
16	SA_CASHOUT_ATTEMPTS	Override number of cashout attempts
17	SA_UNBALANCED_CASHOUT	Unbalanced cashout
18	SA_MANUAL_CANCEL	Manual cancel
19	SA_FORCE_CREDIT	Force credit
20	SA_CANCEL_CASHOUT	Cancel cashout
21	SA_REMOTE_CASHOUT	Remote cashout
22	SA_EVENT_ACTIVATION	Event activation
23	SA_ACS_TICKET_LOOKUP	ACS32 Ticket Lookup
24	SA_RETURN_USED_TICKETS	Return used tickets
25	SA_EXIT_KIOSK	Exit kiosk
26	SA_UNLOAD_SV	Unload stored value
27	SA_STOP_SELF_SERVICE_SCAN	Stop self-service scanning
28	SA_ACS_ABORT_BIOMETRIC_REG	ACS32 abort biometric registration
29	SA_ACS_ABORT_BIOMETRIC_ID	ACS32 abort biometric identification
30	SA_ISTAMP_TOGGLE	I-Stamp toggling
31	SA_ACS_CONSOLE	ACS32 enter console
32	SA_EXIT_GALAXY_FROM_ACS_AT_POS	ACS32 exit Galaxy from ACS at POS
33	SA_PTO	PTO
34	SA_TRAN_REFUND	Transportation refund
35	SA_ACS_CAPTURE_BIOMETRIC	ACS32 capture biometric data
36	SA_EDIT_RATE	Edit rate
37	SA_SMART_CARD_FOP	Smart card FOP
38	SA_ADMIN_AUTH_FUNCTIONS	Administrative authorization functions
39	SA_ALLOW_USAGE_REVERSAL	Allow usage reversal
40	SA_ALLOW_USED_LOCKER_RETURN	Allow return of used lockers
41	SA_EXIT_ACS	Exit ACS32
42	SA_ALLOW_DISABLE_GAK	Allow disabling of ERG GAK
43	SA_ACS_ENABLE_LOGGING	ACS32 enable logging
44	SA_ACS_DISABLE_LOGGING	ACS32 disable logging

45	SA_AC5_TOGGLE_BIOMETRICS	ACS32 toggle biometric scanning
46	SA_AC5_ENABLE_BIOMETRICS	ACS32 enable biometric scanning
47	SA_AC5_DISABLE_BIOMETRICS	ACS32 disable biometric scanning
48	SA_AC5_BIOMETRICS_RATE	ACS32 biometrics rate
49	SA_LOGON_CARD	Logon card
50	SA_LOCK_SCREEN_LOGOFF	Logoff from lock screen
51	SA_CHANGE_ACP_TASK_WITH_DATA	Change ACP task with data
52	SA_CHANGE_ACP_TASK	Change ACP task
53	SA_ALLOW_CONCURRENT_LOGON	Allow concurrent logon
54	SA_DISCOUNT	Discount
55	SA_RS_HOLD_RELEASE	Release RS hold
56	SA_TOGGLE_VEHICLE_DETECTOR_BYPASS	Vehicle detector loop requirement bypass
57	SA_TOGGLE_VEHICLE_DETECTOR_ENABLE	
58	SA_REINITIALIZE_VEHICLE_DETECTOR	
59	SA_PROTOCOL_ADMIN_FUNCTION	
60	SA_PASS_EDIT_EXPIRATION_DATE	Edit pass expiration date
61	SA_PLUGIN_ADMIN_FUNCTION	Plugin Admin Function
62	SA_ITEM_QTY_REDUCTION	
63	SA_CANCEL_A_TRANSACTION	
64	SA_FISCAL_ARCHIVING	
65	SA_ALLOW_REFUND_IN_DOWNGRADE	SA allowed refund during a downgrade configured to not return money.

## 12.2 JnlDetails

The **JnlDetails** table contain records for all financial activity of any type. Because it is an indexed table rather than a flat file, there is a great improvement in reporting. The **JnlDetails** table is the single raw source for financial information and is used in reporting for **IncomeStatements**, etc. Under some configurations, records can be added to this table in a summary mode in which the **AuxTableID** either is not populated or the value is simply informational and does not represent a proven foreign key into an auxiliary table.

At its most basic, the **JnlDetails** table represents a line item view of the transaction, since some rows can represent financial information, and other rows can simply represent additional information regarding the transaction as it appeared in AX. Line numbers are meaningless in the context of the **JnlDetails** table, since either all or none of the details will be written to the table and the **JnlDetailID** represents the sequence in which that row occurred on the transaction.

### Columns

Column	Type	Allow Nulls	Description
JnlDetailID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to <b>JnlHeaders.JnlTranID</b> .
AccountID	Char(12)	N	Pseudo foreign key or shorthand reference to the GLAccount info.
Qty	Integer	N	Quantity involved in transaction.
Amount	Float	N	Money amount involved in transaction.
JnlCodeID	Integer	N	CodeValues.ID for TypeCode <b>JNLCODE</b> . <sup>1</sup>
AuxTableID	Integer	N	Foreign key into the auxiliary table that holds additional information for this entry as specified by the <b>JnlCodeID</b>  OR  integer representation of a value for informational purposes when it is too time-consuming to look up and guarantee a foreign key into another table.
Reference	Char(10)	Y	Provides supplementary information when needed (most commonly used to distinguish between <b>JnlCodeID</b> 101 entries when their <b>AuxTableIDs</b> may refer to rows in the <b>TICKETS</b> table or <b>PASS</b> table as specified in this column).
ActiveInd	Char	N	Active indicator (defaults to Y, may be N if this record is no longer desired to be available for current use, but record is still available for auditing purposes).
CreateDate	Datetime	N	Date this record was created.
CreatedBy	Char(10)	N	The server-provided username of the person who created this record.
OrderLineID	Integer	Y	Foreign key to <b>OrderLines.OrderLineID</b>
PackageDetailID	Integer	Y	The unique ID of the package detail that corresponds to the ticket/item in the transaction
JnlContactID	Int	Y	Foreign key to <b>JnlContacts.JnlContactID</b>

### Indexes

Name	Kind	Columns	Purpose
PKJnlDetailsJnlDetailID	P	JnlDetailID	Primary key (Non-Clustered).
CXJnlDetailsJnlTranIDJnlDtID	CX	JnlTranID, JnlDetailID	Clustered Index  Used to load a list of records from the <b>JnlDetails</b> table where the <b>JnlTranID</b> falls within a specified range.
IXJnlDtlsJnlCdJTDATIDQtyAmt	IX	JnlCodeID, JnlTranID, AuxTableID, Qty, Amount	Used by the reseller reports. Optional index to improve database performance; to add this index run the <b>GalaxyDatabase</b> script with @AddIndexForResellerReports = 1 or @AddIndexesForJNLTables set to 1
IXJnlDetailsJnlAuxTableID	IX	AuxTableID	Improve performance of update queries using <b>AuxTableID</b>
IXJnlDtlsTranIDCodeIDDAmtDtID	IX	JnlTranID, JnlCodeID, Amount, JnlDetailID	Optional index to improve database performance; to add this index run the <b>GalaxyDatabase</b> script with AddIndexesForJNLTables set to 1
IXSQLShiftReport1	IX	JnlCodeID, JnlTranID	Speed up SQL shift reports from Manager's Workstation

<sup>1</sup> **JnlCodeID** Values

Value	Gateway Constant Name	Description	AuxTableID (FK reference to)
4	VOID_REC	Void record	JnlVoid.JnlVoidID
5	REFUND_REC	Refund record	JnlRefunds.JnlRefundID
7	ID_MEMO_REC	ID associated memo record	MemoTbl.MemoTblID
8	POLL_REC	Concentrator poll data indicator	N/A
9	MEMO_REC	General memo record	MemoTbl.MemoTblID
19	CCF_NODE_REC	JnlCCFNode record	JnlCCFNodes.JnlCCFNodeID
20	PACKAGE_DETAIL_REC	Package Detail Record	JnlDetails.PackageDetailID
24	CUST_ID_REC	Customer ID record	Customers.CustomerID
25	ROSTER_REC	Roster record	
26	DISCOUNT_REC	Discount Record	JnlDiscounts.DiscountID
27	OVER_SHORT	Over/short amount for the shift record	JnlOverShort.JnlOverShortID
28	STOCK_ISSUE	Ticket stock issued to booth	JnlStocks.JnlStockID
29	STOCK_USAGE	Ticket stock usage recorded at cashout	JnlStocks.JnlStockID
30	STOCK_RETURN	Ticket stock returned from the booth	JnlStocks.JnlStockID
31	STOCK_VOID	Ticket stock void record	JnlStocks.JnlStockID
32	STOCK_V_RET	Ticket stock returned void record	JnlStocks.JnlStockID
33	CHANGE_REC	Starting change fund for the shift	JnlChangeFunds.JnlChangeFundID
35	DRAFT_REC	Credit authorization information	JnlDrafts.JnlDraftID
37	ORDER_REC	Order transaction record	Orders.OrderID
38	EVENT_REC	Event transaction record	N/A
39	OLD_USAGE_REC	Usage record	N/A
40	INVOICE_REC	Invoice record	ARInvoices.InvoiceID
41	SETTLE_REC	Settlement record	Settlements.SettlementID
42	CURRENCY_REC	Foreign currency record	JnlCurrencies.JnlCurrenctID

43	CAPACITY_REC	Event capacity record	JnlCapacities.JnlCapacityID
44	ASSIGN_SEAT_REC	Assigned seat record for Transportation	JnlAssignSeats.JnlAssignSeatID
45	OLV_DRAFT_REC	Additional draft data	Settlements.SettlementID
46	CCF_MEMO_REC	CCF memo data	JnlCCFMemos.JnlCCFMemoID
47	VISUAL_ID_REC	Visual ID record	N/A
48	RESERVATION_REC	Resource reservation	N/A
49	CUST_ACCOUNT_REC	Customer Account Record	ARAccounts.AccountID
50	NEW_USAGE_REC	New usage record	JnlUsage.JnlUsageID
51	RESELLER_DETAIL_REC	Reseller Information Record	
52	RECEIPT_REC	Receipt printed on demand	JnlReceipts.JnlReceiptID
53	EXTRA_TKT_INFO_REC	Extra ticket information record	N/A
54	REASON_REC	Reason record	JnlReasons.JnlReasonID
55	SURVEY_REC	Survey response record	JnlSurveys.JnlSurveyID
56	DENOMINATION_REC	Denominational deposit record	JnlDenominations.JnlDenominationID
57	SUSPEND_SHIFT_REC	Current shift has been suspended for remote cashout	
58	EXTRA_ITEM_INFO_REC	Transactional tax records for items	N/A
59	CASHOUT_SHIFT_REC		
60	NO_SALE_REC	No sale, open cashdrawer transaction	JnlNoSales.JnlNoSaleID
61	DEBIT_REC	Debit sale transaction	JnlDebits.JnlDebitID
62	RECHARGE_REC	Recharge to debit	JnlRecharges.JnlRechargeID
63	SALES_PROG_REC	Sales program	SalesPrograms.SalesProgramID
64	SV_LOOKUP_REC	Stored value lookup record	JnlSVLookups.JnlSVLookupID
65	EXTRA_DRAFT_REC	Stores Card Verification Number (CVN) and Address Verification Number (AVS) information	JnlDrafts.JnlDraftID
66	UPGRADE_REC	Ticket upgraded record	JnlUpgrades.JnlUpgradeID
67	REPRINT_REC	Ticket reprinted record	JnlReprints.JnlReprintID
68	ENTRY_TAX_REC	Tax info for individual ticket/item  <b>NOTE:</b> This is different than tax record after a payment. This is an entry about the tax related to the ticket or the item journalized	N/A
69	STRATUS_ACTIVATION_REC	Activation of a Stratus gift card	
70	STRATUS_RECHARGE_REC	Recharge of a Stratus gift card	
71	EOT_REC	End Of Transaction	N/A
72	LOYALTY_CARD_REC	Info about loyalty cards and bonus promotions (like Greenbax)	
73	EXTRA_PAID_IO_REC	This record contains the PLU for any Paid In or Paid Out record that is an Orphan Ticket (does not have Product/Level/Fkey defined).	N/A
74	SUPER_TICKET_REC	Super-Ticket information	JnlSuperTickets.JnlSuperTicketID
75	ENCRYPT_REC		
76	ORDER_LINE_REC	Local journal has OrderLine number	N/A
77	DM_MEMO	See JnlHeaders.JnlCode	N/A
78	DM_MEMO_REC	Detail record for editing general config	MemoTbl.MemoTblID
79	BAD_LOGON	See JnlHeaders.JnlCode	
80	BAD_LOGON_MEMO_REC	Detail record for invalid logon	MemoTbl.MemoTblID
81	SKIDATA_SC_ACTIVATION_REC	activations of Skidata Smart Cards	JnlDrafts.JnlDraftID
82	SKIDATA_SC_RECHARGE_REC	recharges of Skidata Smart Cards	JnlDrafts.JnlDraftID
83	PAYMENT_CARD_REC	Payment Card Account Information to store up to 30 digits of an account number	JnlPayments.JnlPaymentID
84	VIEW_LOG_REC	User viewed encrypted (journal) record.	SecurityLog.SecurityLogID
85	GIFT_REC	Transaction includes Item with Register Gift option set. This record is one per Item record	JnlGifts.JnlGiftID
86	UDF_REC	User Defined Field record. This is per transaction and there could be more than one UDF_REC per transaction.	JnlUDFFields.JnlUDFFieldID
87	CONTACT_REC	Contact used for Gift. There is only one CONTACT_REC per transaction	JnlContacts.JnlContactID
88	SALES_PROGRAM_MEMO_REC	Sales Program for Virtual Sales Program	
89	SUPER_LINK_REC	Used for lineage of Super Ticket. When use in Ticket Lookup, this record is use to link all relevant tickets or passes together for purpose of viewing lineage.	N/A JnlSuperLinks.JnlSuperLinkID
90	PTO_REC	Prepurchased Ticket Order for Transportation	JnlPTOs.JnlPTOID
91	TRAN_REFUND_REC	Transportation Refund record	JnlTranRefunds.JnlTranRefundID
92	TRIP_SUMMARY_REC	Transportation Trip Summary Record	JnlTripSummaries.JnlTripSummaryID
93	SURCHARGE_REC	Transportation Surcharges	
94	USAGE2_REC	Extra information related to usage	
95	TRAN_REFUND_REC_2	Transportation Refunds	
96	TRAN_REFUND_REC_3	Transportation Refunds	
97	PACKAGE_HEADER_REC	Mark the beginning of a package	JnlPackages.JnlPackageID
98	PACKAGE_FOOTER_REC	Mark the end of a package	
99	USAGE3_REC	Smartcard usage	
1000	RESERVED_SEAT_REC	Reserved Seating	
1001	CONTACT_NAME_REC	Contact Name	
1002	TRIP_SUMMARY_REC_2	Trip Summary record 2	
1003	TRIP_SUMMARY_REC_3	Trip Summary record 3	

1004	FSFLCT_PLUS_DRAFT_REC_1	eSelectPlusProtocol data	
1005	ESELECT_PLUS_DRAFT_REC_2	eSelectPlusProtocol data	
1006	ESELECT_PLUS_DRAFT_REC_3	eSelectPlusProtocol data	
1007	ESELECT_PLUS_DRAFT_REC_4	eSelectPlusProtocol data	
1008	ESELECT_PLUS_DRAFT_REC_5	eSelectPlusProtocol data	
1009	ESELECT_PLUS_DRAFT_REC_6	eSelectPlusProtocol data	
1010	SIAE_FISCAL_DATA_REC	SIAE Fiscal Data	
1011	SIAE_EVENT_DATA_REC	SIAE Event Data	
1012	SIAE_ITEM_DATA_REC	SIAE Item Data	
1013	SIAE_VENUE_DATA_REC	SIAE Venue Data	
1014	LOYALTY_ACCOUNT_REC	Loyalty Account	
1015	VALUE_LINK_DATA_REC	ValueLink Data	
1016	UPSELL_REC	Upsell data	
1017	SERIAL_NO_USAGE_REC	Serial Number Usage record	
1018	GIFT_AID_REC	Gift Aid record	
1019	TRANS_UPSELL_REC	Transactional upsell record	
1020	SIAE_FISCAL_DATA_2_REC	SIAE additional fiscal data	
1021	SIAE_EVENT_DATA_2_REC	SIAE additional event data	
1022	SIAE_ITEM_DATA_2_REC	SIAE additional item data	
1023	DCC_DRAFT_REC	DCC data	
1024	BUSBILL_REC	Bussbill data	
1025	EXPRESS_SURCHARGE_REC	Express Surcharge data	
1026	TICKET_CONTACT_REC	Ticket Contact data	
1027	MEMBER_ADD_ON_REC	Joint member add-on record	
1028	GUID_REC	GUID support	
1029	BUSBILL_REC2	New busbill data format	
1030	GIFT_AID_REC	Gift Aid record	
1031	TRANS_UPSELL_REC	Transactional upsell record	
1032	SIAE_FISCAL_DATA_2_REC	SIAE additional fiscal data	
1033	EXTRA_DRAFT_2_REC	Additional host data from credit card authorization	
1034	SUPERVISOR_APPROVAL_REC	Header record for a supervisor approval transaction.	
1035	TOGGLE_SURVEY_PROMPTS_REC	Header record for a toggle survey prompts on/off transaction	
1036	TOGGLE_VEHICLE_DETECTOR_BYPASS_REC	Header record for a toggle vehicle detector loop bypass on/off transaction	
1037	TOGGLE_VEHICLE_DETECTOR_ENABLE_REC		
1038	REINITIALIZE_VEHICLE_DETECTOR_REC		
1039	DEVICE_REC		
1040	RFID_DATA_REC	RFID data from RF encoded tickets	
1041	RESELLER_DETAIL_REC_2		
1042	RESELLER_DETAIL_REC_3		
1043	CARD_TOKEN_REC	Host data field record to hold generic host data. Associated with a draft record.	
1044	HOST_DATA_FIELD_REC	Host data field record to hold generic host data. Associated with a draft record.	
1045	MODIFIER_REC	Food and beverage item modifier	
1046	TRANS_HOST_DATA_FIELD_REC	Transaction host data field records to hold generic host data.	JnlHostDataFields.JnlHostDataFieldID
1047	SALES_CHANNEL_REC	Sales channel used in transaction	SalesChannels.SalesChannelID
1048	EXTRA_PAYMENT_REC	Additional payment record information	
1049	MODIFIED_TICKET_DATES_REC	Modified ticket dates information	
1050	DIGITAL_SIGNATURE_REC	Journal transaction digital signature	JnlDigitalSignatures.JnlDigitalSignatureID
1051	FISCAL_SUMMARY_HEADER_REC	Fiscal summary header record	JnlFiscalSummaryHeaders.JnlFiscalSummaryHeaderID
1052	FISCAL_SUMMARY_TAX_DETAIL_REC	Fiscal summary tax details record	JnlFiscalSummaryTaxDetails.JnlFiscalSummaryTaxDetailID
1053	ITEM_QUANTITY_REDUCED_REC	Item quantity reduction record	JnlItemQuantityReductions.JnlItemQuantityReductionID
1054	FISCAL_SUMMARY_DETAIL_REC	Fiscal summary detail record record	JnlFiscalSummaryDetails.JnlFiscalSummaryDetailID
1058	FISCAL_ARCHIVING_REC	Journal fiscal transaction archive record	JnlFiscalTransactionArchiving.JnlFiscalTransactionArchiveld
<b>Journal revenue record codes</b>			
101	TKT_REC	Ticket record	JnlTickets.JnlDetailID
102	ITEM_REC	Item record	JnlItems.JnlItemID
103	FEE_REC	Fee record	JnlItems.JnlItemID
104	DONATION_REC	Donation record	JnlItems.JnlItemID
120	TAX_REC	Tax record	JnlTaxes.JnlTaxesID
210	PAIDIN_REC	Paid-in record	JnlPaidIos.JnlPaidIOID
310	PAIDOUT_REC	Paid-out record	JnlPaidIos.JnlPaidIOID
401	TKT_REFUND_REC	Ticket refund record	N/A
420	ST_TAX_REFUND_REC	Tax refund record	JnlPaidIos.JnlPaidIOID
532	PAYMENT_REC	Payment record	JnlPayments.JnlPaymentID
533	REISSUE_REC	Reissuance information for a transportation ticket.	JnlReissues.JnlReissueID
610	DEPOSIT_REC	Deposit record	JnlDeposits.JnlDepositID

620 COA\_CHANGE\_REC Starting change fund for the shift with COA info JnlChangeFunds.JnlChangeFundID

### 12.3 JnlAssignSeats

The **JnlAssignSeats** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a transportation assigned seat record. The **JnlCodeID** of the corresponding record in the **JnlDetails** table is 44.

#### Columns

Column	Type	Allow Nulls	Description
JnlAssignSeatID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Detail
Name	VarChar(32)	N	Name of the person in the assigned seat
Carrier	Char(4)	N	Bus carrier abbreviation
Sched	Char(4)	N	Bus schedule
Section	Int	N	Bus section
Seat	Char(4)	N	This is the assigned seat number
RecordID	Int	N	Record ID
NumRides	Int	N	Number of Rides
Frequency	Char(8)	Y	Trip frequency
TravelDate	Int	N	Travel date

#### Indexes

Name	Kind	Columns	Purpose
PKJnlAssignSeat	P	JnlAssignSeatID	Primary Key.
CXJnlAssignSeatsTrnIDSeatID		JnlTranID, JnlAssignSeatID	Clustered index. Used to load a list of records from the <b>JnlAssignSeats</b> table where the <b>JnlTranID</b> falls within a specified range

## 12.4 JnlBonuses

The **JnlBonus** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a loyalty card record. Loyalty card records store information for transactions in which a loyalty card was used to issue and/or redeem points. The **JnlCodeID** of the corresponding record in the **JnlDetails** table is 72.

### Columns

Column	Type	Allow Nulls	Description
JnlBonusID	Int	N	Primary key, always unique.
JnlTranID	Int	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
CardInterface	Int	Y	Identifies loyalty card interface used <sup>1</sup> .
Account	VarChar(20)	Y	The loyalty card account number.
PointsIssued	Float	Y	The number of points issued for the transaction.
PromotionalPoints	Float	Y	The number of promotional points issued for bonus promotions.
PointsRedeemed	Float	Y	The number of points redeemed as a form of payment.
IssueRate	Float	Y	The number of points issued per base denomination (dollar).
RedeemRate	Money	Y	The redemption value of each point.
SalesAmount	Money	Y	The transaction amount (excluding tax), used as the basis for computing how many points to issue.

### Indexes

Name	Kind	Columns	Purpose
PKJnlBonusID	P	JnlBonusID	Primary Key.
CXJnlBonusesJnlTrnIDJnlBonusID		JnlTranID, JnlBonusID	Clustered index. Used to load a list of records from the <b>JnlBonuses</b> table where the <b>JnlTranID</b> falls within a specified range.

<sup>1</sup> CardInterface Values

Value	Gateway Constant Name	Description
1	LC_INTERFACE_GREENBAX	Greenbax ( <a href="http://www.greenbax.net">http://www.greenbax.net</a> )

## 12.5 JnlCapacities

The JnlCapacities table contains information from capacity records in the journal.

### Columns

Column	Type	Allow Nulls	Description
JnlCapacityID	Int	N	Unique ID
JnlTranID	Int	Y	Journal transaction ID to tie record to a transaction.
EventID	Int	Y	ID of the Event
ResourceID	Int	Y	ID of the Resource
Quantity	Int	Y	Number of tickets sold in transaction for event.
CapacityID	Int	Y	The unique ID of the related RMCapacity record

### Indexes

Name	Kind	Columns	Purpose
PKJnlCapacitiesJnlCapacityID	PK	JnlCapacityID	Index on Unique ID
CXJnlCapacities	CX	JnlTranID, JnlCapacityID	Reporting

## 12.6 JnlCashoutShifts

This is an auxiliary journal table and will store the values from the CASHOUT\_SHIFT\_REC journal record. This type of record is journalized in a cashout transaction when the user executes a remote cashout. For each shift been cashed out in a remote cashout, one CASHOUT\_SHIFT\_REC record will be journalized. Then, if the remote cashout includes 3 different shifts, 3 CASHOUT\_SHIFT\_REC records will be generated.

### Columns

Column	Type	Allow Nulls	Description
JnlCashoutShiftID	Int	N	Unique. Primary Key. Generated from GatewayCounters.
JnlTranID	Int	N	References JnlHeader.JnlTranID
NodeNo	Int	N	Node where the shift been cashed out were registered
ShiftNo	Int	N	Shift no of the shift been cashed out. (Attention: the number maybe of a shift executed in another machine as we are talking about remote cashout.)
FirstTranNo	Int	N	First Transaction from the shift been cashed out. (Attention: the transaction number is from the NodeNo/ShiftNo shift. It is not necessarily from the current shift in the current machine). This is journalized in order to allow DBSynch to identify a specific shift record in the ShiftStatus table.
ShiftStatus	Int	N	This reflects the status of that shift at the cashout time and based on Journal SQL tables. Either if the status changes after the remote cashout is executed, this will still reflects the status at the cashout time. <sup>1</sup>
RmtCashoutID	Int	N	All the shifts cashed out together in a remote cashout have the same remote cashout id. This is journalized in order to allow DBSynch to rebuild the ShiftStatus table, RemoteCashoutId field.

### Indexes

Name	Kind	Columns	Purpose
PKJnlCashoutShiftID	P	JnlCashoutShiftID	Primary key.
CXJnlCshfstsJnlTrnIDJnlCSID		JnlTranID, JnlCashoutShiftID	Clustered Index.  Used to load a list of records from the JnlCashoutShifts table where the JnlTranID falls within a specified range.

<sup>1</sup> ShiftStatus Values

Value	Gateway Constant Name	Description
1	OPENED_SHIFT	Means that the SUSPEND or CASHOUT record of that shift has not been DBSynched yet. Or maybe the cast member has not suspended or cashed out the shift yet. The shift is still considered opened at this time.
2	SUSPENDED_SHIFT	The cast member has suspended the shift and the SUSPEND journal record has already been DBSynched at the cashout time.
3	CASHED_OUT_SHIFT	The current shift in the node were the remote cashout is been executed is always included in the remote cashout. This shift is been just closed at the cashout time, then its status is displayed as CURRENT.

## 12.7 JnlCCFTables

List of all the tables that have been included for a given CCF.

### Columns

Column	Type	Allow Nulls	Description
JnlCCFTableID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Details
TableID <sup>1</sup>	Int	N	Numeric ID of the table selected
TableAction <sup>2</sup>	Int	N	Add, Delete, Revise, or Overwrite
FileFormat <sup>3</sup>	Int	N	CCFLegacy, CSV. Format the exported data is in.

### Indexes

Name	Kind	Columns	Purpose
PKJnlCCFTablesCCFTableID	P	JnlCCFTableID	Primary Key.

#### <sup>1</sup> Table ID Values

See Table ID's as Indexed at the beginning of this document.

#### <sup>2</sup> Table Action Values

Value	Gateway Constant Name	Description
0	NONE	None
1	ACTION_ADD	Add
2	ACTION_REVISE	Revise
3	ACTION_DELETE	Delete
4	ACTION_OVERWRITE	Overwrite

#### <sup>3</sup> File Format Values

Value	Gateway Constant Name	Description
1	LEGACYCCF	Original CCF format
2	CSV	CSV
3	XML	XML
4	SQL_BULK	SQL Bulk Insert

## 12.8 JnlCCFMemos

The **JnlCCFMemos** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a CCFMemo. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 46.

### Columns

Column	Type	Allow Nulls	Description
JnlCCFMemoID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
CCFMemoType	Char(8)	N	Category/type of CCF memo. <sup>1</sup>
CCFMemoLow	Integer	N	First line number of described portion of CCF file.
CCFMemoHigh	Integer	N	Last line number of described portion of CCF file.
CCFMemoFile	Char(12)	N	Filename of the CCF.
MemoTblID	Integer	N	Always contains 0. This column is not currently used by the system.

### Indexes

Name	Kind	Columns	Purpose
PKJnlCCFMemosJnlCCFMemoID	P	JnlCCFMemoID	Primary key.
CXJnlCCFMmosJnlTrnIDJnlCCFMmID		JnlTranID, JnlCCFMemoID	Clustered Index.  Used to load a list of records from the <b>JnlCCFMemos</b> table where the <b>JnlTranID</b> falls within a specified range.

<sup>1</sup> **CCFMemoType**

Value	Gateway Constant Name	Description
CCF1	CCF_MEMO_SKIP	For records in a CCF that were skiped, usually due to some records attempting to update a database that is not enabled.
CCF2	CCF_MEMO_ABORT	For CCFs which could not be processed.
CCF3	CCF_MEMO_IMPORT	Marks completion of CCF apply process.
CCF4	CCF_MEMO_EXPORT	Marks completion of CCF creation.

## 12.9 JnlCCFNodes

Records and tracks the status of each node that is to receive the CCF.

### Columns

Column	Type	Allow Nulls	Description
JnlCCFNodeID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Details
Node	Int	N	Node number
Status <sup>1</sup>	Int	N	Current status - Waiting, Received CCF, Processed CCF, Error
StatusMessage	Char(30)	N	System Error Message
FileName	Char(30)	N	Filename of the CCF

### Indexes

Name	Kind	Columns	Purpose
PKJnlCCFNodesCCFNodeID	P	JnlCCFNodeID	Primary Key.

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
NODE_WAITING	01	Waiting to receive CCF
NODE_SENT	02	CCF has been sent to the node
NODE_RECEIVED	03	Received CCF
NODE_PROCESSED	04	CCF has been processed
NODE_ERROR	05	An error has occurred. See Status Message for details.

## 12.10 JnlCCFStatuses

This table is used to track the current status of a given CCF. The data provided here can be viewed via a CCF maintenance facility.

### Columns

Column	Type	Allow Nulls	Description
JnlCCFStatusID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link back to the Journal Details
FileName	Char(30)	N	CCF Filename
EffectiveDate	DateTime	N	Date/Time CCF is to be applied
SentDate	DateTime	Y	Date/Time CCF was sent to all the nodes
Status <sup>1</sup>	Int	N	Status code of the CCF

### Indexes

Name	Kind	Columns	Purpose
PKJnlCCFStatusesCCFStatusID	P	JnlCCFStatusID	Primary Key.

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	STATUS_SEND	CCF has been created but not sent out.
2	STATUS_SENT	CCF has been sent to all the nodes
3	STATUS_COMPLETE	CCF has been created, sent, received, and processed by all nodes.

## 12.11 JnlChangeFunds

This table contains the change fund journalized by the system at the beginning of the shift.

### Columns

Column	Type	Allow Nulls	Description
JnlChangeFundID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key to JnlHeaders.JnlTranID
Amount	Money	N	The starting change fund for the shift.
FOPID	Int	N	The FOP to apply the change fund to. Foreign key reference to FOPs.FOPID.
CashBagNumber	Int	Y	Cashbag Number
ChangeFundDescription	Char(20)	Y	Anything the user wants to store - Free form field
AdditionalChangeFund	Bit	Y	Set to True if this Change Fund is NOT the start of shift change fund

### Indexes

Name	Kind	Columns	Purpose
PKJnlChangeFundsChangeFundID	P	JnlChangFundID	Primary key.
CXJnlChangeFnDsJnlTrnIDJnlCFID		JnlTranID, JnlChangeFundID	Clustered Index.  Used to load a list of records from the JnlChangeFunds table where the JnlTranID falls within a specified range.

## 12.12 JnlContacts

This table contains journal records linking a contact to other journalized activity or purchase.

### Columns

Column	Type	Allow Nulls	Description
JnlContactID	Int	N	Primary key, always unique
JnlTranID	Int	N	Foreign key to JnlHeaders.JnlTranID.
ContactID	Int	Y	Foreign key to CustContacts.CustContactID.
ExternalID	VarChar(40)	Y	Another ID that may be used to refer to the contact.
ExternalTypeID	Int	Y	This field will indicate the type of number that is provided in the ExternalID field. If a contact is selected only, the ExternalID and this field will be blank.
FirstName	VarChar(30)	Y	First name of the individual
LastName	VarChar(30)	Y	Last name of the individual

### Indexes

Name	Kind	Columns	Purpose
PKJnlContactsJnlContactID	PK	JnlContactID	Primary Key.
CXJnlContactsJnlTranIDJnlContID	CX	JnlTranID	Clustered index on JnlTranID
IJnlContactIDContactID		JnlContactID, ContactID	Used when querying for spending history for a Contact

## 12.13 JnlCurrencies

This table contains Journal currency information. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 42.

### Columns

Column	Type	Allow Nulls	Description
JnlCurrcyID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key to JnlHeaders.JnlTranID.
ExchRate	Float	N	Exchange rate for the foreign currency.
CurrencyKey	Char(1)	N	Value ("A" through "Z") indicating the type of currency.

### Indexes

Name	Kind	Columns	Purpose
PKJnlCurrenciesJnlCurrcyID	P	JnlCurrcyID	Primary key.
CXJnlCurrencsJnlTrnIDJnlCurrlD		JnlTranID, JnlCurrcyID	Clustered Index. Used to load a list of records from the JnlCurrencies table where the JnlTranID falls within a specified range.

## 12.14 JnlDCCDrafts

The **JnlDCCDrafts** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a DCC Draft record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 1023. A DCC Draft record will exist in the journal when the Dynamic Currency Conversion feature is utilized with Paymentech/NetConnect.

### Columns

Column	Type	Allow Nulls	Description
JnlDCCDraftID	Int	N	Primary key, always unique
JnlTranID	Int	N	Foreign key to <b>JnlHeaders.JnlTranID</b>
RequestIndicator	Int	N	Request indicator as returned by the Fexco service <sup>1</sup>
BaseAmount	Money	N	Amount in base currency for the authorization
ExchangeAmount	VarChar(12)	N	Amount of sale in the international currency
DecimalPlaces	Int	N	Number of decimal places for international currency, as defined by the ISO 4217 standard based on <b>DCCCurrencyCodeID</b>
ExchangeRate	VarChar(13)	N	Exchange rate used to calculate <b>DCCAmount</b>
CurrencyCodeID	Int	N	Currency Code ID used for conversion, as defined by the ISO 4217 standard
CurrencyCode	Char(3)	N	Three-character currency abbreviation as defined by the ISO 4217 standard
FOPNo	Int	N	Form of payment used for the authorization, FK to <b>FOPs.FOPCode</b>

### Indexes

Name	Kind	Columns	Purpose
PKJnlDCCDraftID	P	JnlDCCDraftID	Primary Key
CXJnlDCCDraftsTrnIDDDCCDraftID		JnlTranID, JnlDCCDraftID	Clustered Index.  Used to load a list of records from the <b>JnlDCCDrafts</b> table where the <b>JnlTranID</b> falls within a specified range.

<sup>1</sup> RequestIndicator Values

Value	Gateway Constant Name	Description
0	DCC_ACCEPTED	Transaction qualified for DCC processing and accompanying DCC data is present
1	DCC_NOT_APPLICABLE	N/A (not applicable in HCS)
2	DCC_DECLINED	DCC declined by cardholder
3	DCC_TERMINAL_DISABLED	Terminal disables DCC
4	DCC_NOT_REQUESTED	Terminal is DCC capable, but not requesting

## 12.15 JnlDebits

The **JnlDebits** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a debit record. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 61. A transaction will consist of one debit record for each ticket record that was a debit card sale.

### Columns

Column	Type	Allow Nulls	Description
JnlDebitID	Int	N	Primary key, always unique.
JnlTranID	Int	Y	Foreign key to JnlHeaders.JnlTranID.
DebitTypeID	Int	Y	Foreign key to DebitTypes.DebitTypeID.
ExpirationDate	Datetime	Y	The date the debit card expires; zero indicates no expiration date.
VisualID	Varchar(60)	Y	VisualID of stored value card
EntryMethod	Int	Y	Indicates how the card number was entered (i.e., swipe, manual, etc)
DataConnection	Int	Y	Indicates which data connection was used to process the transaction
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the debit card number (VisualID).
HostData	Varchar(255)	Y	Used to hold data journalized by the host system that may be needed in a void of a Debit/Draft/Recharge.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDebitsJnlDebitID	P	JnlDebitID	Primary Key.
CXJnlDebitsJnlTranIDJnlDebitID		JnlTranID, JnlDebitID	Clustered Index.  Used to load a list of records from the JnlDebits table where the JnlTranID falls within a specified range.

## 12.16 JnlDenominations

The **JnlDenominations** table provides denominational deposit information. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 56.

### Columns

Column	Type	Allow Nulls	Description
JnlDenominationID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID.
DenomID	Integer	Y	Foreign key to Denominations.DenomID.
CurrencyAbbr	Char(3)	Y	Currency abbreviation of the denomination deposited.
Qty	Integer	Y	Quantity of the denomination deposited.
ExchangeRate	Float	Y	Exchange rate from the deposit currency to Galaxy's base currency.
Value	Float	Y	Amount converted to the Galaxy's base currency.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDenominationsJnlDenomID	P	JnlDenominationID	Primary Key.
CXJnlDenomsJnlTrnIDJnlDenomID		JnlTranID, JnlDenominationID	Clustered Index.  Used to load a list of records from the JnlDenominations table where the JnlTranID falls within a specified range.

## 12.17 JnlDeposits

The **JnlDeposits** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a deposit. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 610.

### Columns

Column	Type	Allow Nulls	Description
JnlDepositID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
ExchAmount	Float	N	Amount of money in deposited currency.
CurrencyKey	Char	N	Value ("A" through "Z") indicating the type of currency.
DepositRef	Char(8)	N	Deposit ID entered at the POS to identify the deposit.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDepositsJnlDepositID	P	JnlDepositID	Primary key.
CXJnlDepositsJnlTrnIDJnlDpstID		JnlTranID, JnlDepositID	Clustered Index.  Used to load a list of records from the <b>JnlDeposits</b> table where the <b>JnlTranID</b> falls within a specified range.
IXJnlDepositsCurrKeyTranIDDepID	IX	CurrencyKey, JnlTranID, JnlDepositID	Optional index to improve database performance; to add this index run the <b>GalaxyDatabase</b> script with <b>@AddIndexesForJNLTables</b> set to 1

## 12.18 JnlDigitalSignatures

The JnlDigitalSignatures table provides supplementary information for the corresponding JnlDetails record when that detail refers to a digital signature. The JnlCodeID value of the corresponding record in the JnlDetails table is 1050.

### Columns

Column	Type	Allow Nulls	Description
JnlDigitalSignatureID	Int	N	Primary key, always unique, Identity.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
SignatureVersion	Int	N	A version number used to support backwards compatibility for digital signature formats.
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the signature.
Signature	NVarchar(MAX)	Y	The encrypted digital signature value.
SourceNodeNo	Int	N	The node that generated the signature or is responsible for generating the signature if pending.
Pending	Bit	Y	Indicates if the signature is pending. SourceNodeNo will indicate which node is responsible for generating the signature.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDigitalSignatureID	P	JnlDigitalSignatureID	Primary Key.
CXJnlDigitalSignatures		JnlTranID, JnlDigitalSignatureID	Clustered Index.  Used to load a list of records from the JnlDigitalSignatures table where the JnlTranID falls within a specified range.
IXJnlDigitalSignaturesPending		Pending	Used for finding all signatures that are still pending.
IXJnlDigitalSignaturesSourceNodeNo		SourceNodeNo	Used for finding the pending digital signatures by the node responsible for generating them.

## 12.19 JnlDiscounts

The JnlDiscounts table stores all the applicable discount records from JNL.DAT for each ticket in the JnlTickets table. A single JnlTickets records can have one to many records in this table, each record representing a discount applied to the ticket. This table is populated by DBSync when exporting the journal to SQL.

### Columns

Column	Type	Allow Nulls	Description
JnlDiscountID	Int	N	Primary key, always unique
JnlTranID	Int	N	Foreign key to the JnlHeaders.JnlTranID
JnlTicketID	Int	N	Foreign key to JnlTickets.JnlDetailID
DiscountAmount	Money	N	Amount of discount
DiscountKind	Int	N	Type of discount <sup>1</sup>
DiscountID	Int	N	Foreign key to the applied discount, depends on DiscountType. Could be a POS Discount ID, Sales Program ID.
JnlItemID	Int	Y	Foreign key to JnlItems.JnlItemID
SupervisorID	Int	Y	ID of the user who gave approval
DiscountCode	nvarchar(30)	Y	A unique alphanumeric code for the discount.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDiscountsJnlDiscountID	P	JnlDiscountID	Primary Key
IXJnlDiscountsJnlTicketID		JnlTicketID	Speed up looking up by JnlTicketID

### <sup>1</sup> DiscountKind Values

Value	Gateway Constant Name	Description
0	DISCOUNT_KIND_PRICE_OVERRIDE	User overrode the price of the ticket
1	DISCOUNT_KIND_SALES_PROGRAM	Sales program changed the price of the ticket or item
2	DISCOUNT_KIND_COUPON_DISCOUNT	A POS/Coupon discount was applied to the ticket or item
3	DISCOUNT_KIND_ITEM_DISCOUNT	Item discount was applied
4	DISCOUNT_KIND_LINE_ITEM_DISCOUNT	Line item discount was applied

## 12.20 JnlDrafts

The **JnlDrafts** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a draft record. Draft records are journalized whenever a form of payment is authorized (credit cards, for example). The JnlCodeID value of the corresponding record in the **JnlDetails** table is 35.

### Columns

Column	Type	Allow Nulls	Description
JnlDraftID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to JnlHeaders.JnlTranID.
JnlCodeID	Integer	N	Always 35 (matches JnlCodeID of corresponding record in the JnlDetails table).
FOP	Integer	N	Foreign key to FOPs.FOPID, specifying the form of payment that was authorized.
CardNo	VarChar(50)	N	The account number that was authorized.
ExprDate	Char(4)	N	Expiration date.
AuthCode	VarChar(128)	N	Approval Code (returned from host). <sup>1</sup>
TransID	Char(15)	N	Transaction ID (returned from host). <sup>1</sup>
Validation	Char(4)	N	Validation Code (returned from host). <sup>1</sup>
PSI	Char(1)	N	Payment Service Indicator (returned from host). <sup>1</sup>
AuthSource	Integer	N	Indicates source of approval code. <sup>2</sup>
AccountSource	Integer	N	Indicates how the account number was entered into the system. <sup>3</sup>
Captured	Bit	N	True if payment captured by host. <sup>1</sup>
ReceiptNo	Integer	N	Receipt number (returned from host). <sup>1</sup>
CVNPresenceIndicator	Char(1)	Y	Indicates whether or not a card verification number (CVN) was sent to the host. <sup>4</sup>
CVNResult	Char(1)	Y	Result of CVV2/CVC2/CID card verification if CVN was used (returned from host). <sup>5</sup>
AVSResult	Char(5)	Y	Result of address verification if AVS was used (returned from host). <sup>1</sup>
TraceNumber	Char(8)	Y	Trace number (returned from host). <sup>1</sup> Not supported for many protocols, including some where the host returns values they refer to as "trace numbers".
SignatureIndex	Int	Y	A non-zero value indicates that a signature has been captured.  A value of 0 indicates that no signature was captured.  Value is the number of the payment (chronologically) in a transaction that the captured signature is associated with. This number corresponds to the FOPSequence value in the Signatures table.
GxKeyID	Int	Y	The ID that corresponds to the Key which will decode the encoded Credit Card number, saved in the CardNo column.
DebitAccount	Char(1)	Y	Type of Interac account used <sup>6</sup>
LanguageCode	Integer	Y	Language code of Canadian Interac debit card holder <sup>7</sup>
TransID2	VarChar(20)	Y	The unique transaction identifier from Protocol Host system. Format is TPTYYYYMMDDHHMMSSNNN where TPT is the system
VoidTransID	VarChar(20)	Y	The unique Void Transaction identifier from the Host system
RedeemedValue	Float	Y	Total points redeemed by the guest for the sales transaction
HostTransDateTime	DateTime	Y	Date and time of the transaction
CardType	Char(1)	Y	Identifies the card type used <sup>8</sup>
HostData	Varchar(255)	Y	Used to hold data journalized by the host system that may be needed in a void of a Debit/Draft/Recharge.
HostField1	NVarChar(20)	Y	Holds field data from a credit card host system that must be journalized.
HostField2	NVarChar(30)	Y	Holds field data from a credit card host system that must be journalized.
CardToken	NVarChar(62)	Y	Holds card token data returned from a credit card host system. This value will be encrypted using the GxKeyID in the JnlDrafts if the form of payment is configured to encrypt.
Bin	Char(6)	Y	Contains the BIN (the first 6 digits) of the card number used for the payment, if the "Collect Endorsement BINs" configuration option is turned on.
OfflineCardTokenRetrieved	Bit	Y	When set to 1, indicates that this JnlDraft record originally was created when the payment processor was offline, and the record was later updated with the card token details when the payment processor came back online.
OfflineCardTokenRetrievalDate	DateTime	Y	Indicates the date when this record was updated with the card token details.

### Indexes

Name	Kind	Columns	Purpose
PKJnlDraftsJnlDraftID	P	JnlDraftID	Primary Key.
CXJnlDraftsJnlTranIDJnlDraftID		JnlTranID, JnlDraftID	Clustered Index.  Used to load a list of records from the JnlDrafts table where the JnlTranID falls within a specified range.
IXJnlDraftsCardToken	AK	CardToken	Improve load performance when loading drafts by card token.
IXJnlDraftsCardNo	AK	CardNo	Improve load performance when loading drafts by card number.

<sup>1</sup> Note: Depending upon the credit card protocol being used, the format of these fields can vary. For certain protocols, some of these fields may always be empty or zero.

<sup>2</sup> AuthSource Values

Value	Gateway Constant Name	Description
0	AUTH_SOURCE_BLANK	Not a credit/debit card transaction.
1	AUTH_SOURCE_ONLINE	Authorization was received electronically.
2	AUTH_SOURCE_REFERRAL	Transaction was referred, approval code entered during referral process.
3	AUTH_SOURCE_OFFLINE	Transaction was declined, user entered approval code.

**<sup>3</sup> AccountSource Values**

Value	Gateway Constant Name	Description
0	ENDORSEMENT_BLANK	Not a credit/debit card transaction, for example an adjustment.
1	ENDORSEMENT_SWIPED	The account number was read from a magnetic swipe reader.
2	ENDORSEMENT_MANUAL	The account number was typed in at the endorsement prompt.
3	ENDORSEMENT_SCANNED	The account number was scanned at the endorsement prompt.
4	ENDORSEMENT_CONTACTLESS	The account number was collected by contactless method.
5	ENDORSEMENT_CHIP	The account number was collected using an EMV chip.
6	ENDORSEMENT_FSWIPE	The account number was swiped due to a chip failure.

**<sup>4</sup> CVNPresenceIndicator Values**

Value	Description
0	A CVN was not sent to the host.
1	A CVN was sent to the host.
2	The verification number on the card was not legible (not supported for some protocols).
9	The verification number is not on the card (not supported for some protocols).

**<sup>5</sup> CVNResult Values**

Value	Description
M	The verification was successful.
N	The verification was not successful, due to an incorrect verification value. This is an indication of possible fraud.
P	The host did not process the CVN verification.
S	Although the user indicated that a verification number was not on the card, the host replied that it should have been.
U	The issuer is unable to support CVN.
(space)	Either a CVN was not sent to the host, or the host did not return a CVN result (for some protocols, the host never returns a CVN result).

**<sup>6</sup> DebitAccount Values**

Value	Gateway Constant Name	Description
(space)		Not an Interac transaction
1	ACCOUNT_CHECKING	Checking account
2	ACCOUNT_SAVINGS	Savings account
3	NETS_CUPS	NETS protocol purchase/debit/CUPS
4	NETS_CASHCARD	NETS protocol CashCard purchase (void not allowed)
5	NETS_CONTACTLESS_DEBIT	NETS contactless debit purchase (void not allowed)

**<sup>7</sup> LanguageCode Values**

Value	Gateway Constant Name	Description
0	UNKNOWN_LANGUAGE-255	Not an Interac transaction
1	ENGLISH_LANGUAGE+1	Interac cardholder has English language preference
2	FRENCH_LANGUAGE+1	Interac cardholder has French language preference

**<sup>8</sup> CardType Values**

Value	Description
V	Visa
M	Mastercard
A	American Express
J	JCB
D	Diners
C	Cup
E	EZLink

## 12.21 JnleSelectPlusDrafts

This table is populated when performing transactions in Galaxy with the eSelect PLUS credit card protocol. Fields returned by the eSelect PLUS system in a TerminalReceipt structure are recorded in this table without translation or modification by Galaxy in most cases.

### Columns

Column	Type	Allow Nulls	Description
JnleSelectPlusDraftID	Int	N	Primary Key
JnlTranID	Int	Y	Journal Transaction ID
AccountType	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
Aid	Char(32)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
Amount	Float	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
AppLabel	Char(16)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
AppPreferredName	Char(16)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
ARQC	Char(16)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
AuthCode	VarChar(128)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
CardPlan	Char(16)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
CardType	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
CvmlIndicator	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
ECom	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
ErrorCode	Int	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
ErrorMessage	Varchar(57)	Y	The first 57 characters of the corresponding field in eSelect PLUS TerminalReceipt with exception of ErrorCode = 0, in which case this field is NULL or empty.
IsoCode	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
Lang	Int	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
MCEmvData1	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
MCEmvData2	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
MCEmvData3	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
OrderID	Char(20)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
Pan	Char(4)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt
PanEntry	Char(2)	Y	Maps directly to corresponding field in eSelect PLUS TerminalReceipt

### Indexes

Name	Kind	Columns	Purpose
PKJnleSelectPlusDraftID	P	JnleSelectPlusDraftID	Primary Key (non-clustered).

## 12.22 JnlExpressBusbills

The **JnlExpressBusbills** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a busbill record. The **JnlCodeID** of the corresponding record in the **JnlDetails** table is 1024.

### Columns

Column	Type	Allow Nulls	Description
JnlExpressBusbillID	Int	N	Primary key, always unique
JnlTranID	Int	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
BusbillNumber	NVarChar(14)	Y	FK to <b>ExpressBusbills.BusbillNumber</b>
TransactionMode	Int	Y	Busbill Transaction Mode <sup>1</sup>
BatchEntered	Bit	Y	Defines whether or not this busbill was entered as part of a batch.
BusbillGUID	UniqueIdentifier	Y	Used to uniquely associate an express busbill journal entry with an express busbill in the case of duplicate Express busbill numbers.

### Indexes

Name	Kind	Columns	Purpose
PKJnlExpressBusbillID	P	JnlExpressBusbillID	Primary Key.
CXJnlExpressBusbills	A	JnlTranID, JnlExpressBusbillID	Clustered index. Used to load a list of records from the <b>JnlExpressBusbills</b> table where the <b>JnlTranID</b> falls within a specified range.
IXBusbillGUIDTranIDMode	A	BusbillGUID, JnlTranID, TransactionMode	For express reports

<sup>1</sup> TransactionMode Values

Value	Gateway Constant Name	Description
1	EXPRESS_FORWARD	Forward transaction
2	EXPRESS_RECEIVE	Receive transaction

## 12.23 JnlExpressSurcharges

The **JnlExpressSurcharges** table provides surcharge information for a transaction. The **JnlCodeID** of the corresponding record in the **JnlDetails** table is 1025.

### Columns

Column	Type	Allow Nulls	Description
JnlExpressSurchargeID	Int	N	Primary key, always unique
JnlTranID	Int	Y	Foreign key to JnlHeaders.JnlTranID.
ChargeName	varchar(10)	Y	Name of the surcharge
ChargePLU	char(20)	Y	PLU associated to the surcharge
SurchargeKey	varchar(16)	Y	Foreign Key to ExpressBusbillSurcharges.SurchargeKey
SurchargeAmount	Money	Y	Amount of the surcharge
Amount	Money	Y	Amount of the surcharge
Basis	Integer	Y	Basis of the Surcharge <sup>1</sup>
Method	Integer	Y	Method of the Surcharge <sup>2</sup>
AppliesTo	Integer	Y	What the surcharge applies to <sup>3</sup>

### Indexes

Name	Kind	Columns	Purpose
PKJnlExpressSurchargeID	P	JnlExpressSurchargeID	Primary Key.
CXJnlExpressSurcharges	A	JnlTranID, JnlExpressSurchargeID	Clustered index. Used to load a list of records from the JnlExpressSurcharges table where the JnlTranID falls within a specified range.

#### <sup>1</sup> Basis Values

Value	Gateway Constant Name	Description
0	TABLE_BASIS_VALUE	Value is the actual value
1	TABLE_BASIS_RATE	Value is a rate
2	TABLE_BASIS_PERCENT	Value is a percentage

#### <sup>2</sup> Method Values

Value	Gateway Constant Name	Description
0	SURCHARGE_METHOD_ADD	Add the surcharge to the existing charge.
1	SURCHARGE_METHOD_REPLACE	Replace the current charge with the surcharge value.
2	SURCHARGE_METHOD_ADDNOTZERO	Add the surcharge to the existing charge if the existing charge is not 0.
3	SURCHARGE_METHOD_REPLACEZERO	Replace the current charge with the surcharge value if the charge value is not 0.

#### <sup>3</sup> AppliesTo Values

Value	Gateway Constant Name	Description
0	SURCHARGE_APPLIES_TO_ORIGINAL	The surcharge is applied to the original charge amount.
1	SURCHARGE_APPLIES_TO_NEW	The surcharge is applied to the new charge amount (the original charge amount plus any surcharges applied).

## 12.24 JnlFiscalSummaryDetails

The JnlFiscalSummaryDetails table provides supplementary information for the corresponding JnlDetails record when that detail refers to a fiscal summary detail. The JnlCodeID value of the corresponding record in the JnlDetails table is 1054.

### Columns

Column	Type	Allow Nulls	Description
JnlFiscalSummaryDetailID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
SummaryDetailType	int	N	0 = Period, 1 = Year to date, 2 = Start of Node
SalesTotal	money	N	Total sales for the SummaryDetailType
NonTaxedSalesTotal	money	N	Total amount of non-taxed sales for the SummaryDetailType
TaxTotal	money	N	Total amount of tax collected for the SummaryDetailType
PaymentsTotal	money	N	Total amount of Payments collected for the SummaryDetailType

### Indexes

Name	Kind	Columns	Purpose
PKJnlFiscalSummaryDetailsID	P	JnlFiscalSummaryDetailsID	Primary Key.
CXJnlFiscalSummaryDetails		JnlTranID, JnlFiscalSummaryDetailsID	Clustered Index.  Used to load a list of records from the JnlFiscalSummaryDetails table where the JnlTranID falls within a specified range.

## 12.25 JnlFiscalSummaryHeaders

The JnlFiscalSummaryHeaders table provides supplementary information for the corresponding JnlDetails record when that detail refers to a fiscal summary header. The JnlCodeID value of the corresponding record in the JnlDetails table is 1051.

### Columns

Column	Type	Allow Nulls	Description
JnlFiscalSummaryHeaderID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
PeriodType	int	N	0 = Daily, 1 = Monthly, 2 = Yearly
RollupDate	datetime	N	Date the rollup was performed for.
FiscalID	nvarchar(20)	Y	FiscalID of the company for the transaction.

### Indexes

Name	Kind	Columns	Purpose
PKJnlFiscalSummaryHeaderID	P	JnlFiscalSummaryHeaderID	Primary Key.
CXJnlFiscalSummaryHeaders		JnlTranID, JnlFiscalSummaryHeaderID	Used to load a list of records from the JnlFiscalSummaryHeaders table where the JnlTranID falls within a specified range.

## 12.26 JnlFiscalSummaryTaxDetails

The JnlFiscalSummaryTaxDetails table provides supplementary information for the corresponding JnlDetails record when that detail refers to a fiscal summary tax detail. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 1052.

### Columns

Column	Type	Allow Nulls	Description
JnlFiscalSummaryTaxDetailID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
Category	int	N	Number of the tax for this detail. (1 - 8)
TaxName	nvarchar(8)	N	Name of the tax.
TaxableSalesAmount	money	N	Total of all items sold for this tax category.
TaxCollectedAmount	money	N	Total amount of taxes collected for this tax category.
TaxRate	float	N	Tax rate for this tax category.

### Indexes

Name	Kind	Columns	Purpose
PKJnlFiscalSummaryTaxDetailID	P	JnlFiscalSummaryTaxDetailID	Primary Key.
CXJnlFiscalSummaryTaxDetail		JnlTranID, JnlFiscalSummaryTaxDetailID	Clustered Index.  Used to load a list of records from the JnlFiscalSummaryTaxDetails table where the JnlTranID falls within a specified range.

## 12.27 JnlFiscalTransactionArchiving

The JnlFiscalTransactionArchiving table provides supplementary information for the corresponding JnlDetails record. It contains the dates from and thru that were selected by the operator when they performed a fiscal transaction archive. It also tells us who performed the archiving function and when they performed it. The JnlCodeID value of the corresponding record in the JnlDetails table is 1058.

### Columns

Column	Type	Allow Nulls	Description
JnlFiscalTransactionArchiveID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
FromDate	datetime	N	From Date selected by the operator when performing the archive
ThruDate	datetime	N	Thru Date selected by the operator when performing the archive
ArchiveGUID	uniqueidentifier	Y	Unique identifier for the archive operation.
ArchiveFileHash	nvarchar(max)	Y	The cryptographic hash of the output archive file generated at the time of creation.

### Indexes

Name	Kind	Columns	Purpose
PKJnlFiscalTransactionArchiveID	P	JnlFiscalTransactionArchiveID	Primary Key.
CXJnlFiscalTransactionArchiveID		JnlTranID, JnlFiscalTransactionArchiveID	Clustered Index.  Used to load a list of records from the JnlFiscalTransactionArchiving table where the JnlTranID falls within a specified range.

**12.28 JnlGifts**

This table contains gift information for PLU. This is only journalized if PLU has RegisterGift option set and a Contact is selected for the transaction.

**Columns**

Column	Type	Allow Nulls	Description
JnlGiftID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID.
CampaignID	Integer	Y	Foreign key to Campaigns.CampaignID
FundID	Integer	Y	Foreign key to Funds.FundID
AppealID	Integer	Y	Foreign key to Appeals.AppealID
SolicitationID	Integer	Y	Foreign key to Solicitations.SolicitationID

**Indexes**

Name	Kind	Columns	Purpose
PKGiftID	P	JnlGift	Primary Key (non-clustered).
CXJnlGiftsJnlGiftID		JnlTranID, JnlGiftID	Clustered Index. Used to load a list of records from the JnlGifts table where the JnlTranID falls within a specified range.

## 12.29 JnlHostDataFields

This table stores generic payment host data field information that does not fit in the draft or extra draft records.

### Columns

Column	Type	Allow Nulls	Description
JnlHostDataFieldID	Integer	N	Primary key, always unique.
JnlDraftID	Integer	N	Foreign key to JnlDrafts.JnlDraftID. Payment draft that this host field is associated with.
FieldID	Integer	N	Numerical identifier for the type of field data contained by this record. Values 1-99 are reserved for generic data needed for the protocol, and values 100 and up are for known types of data. <sup>1</sup>
FieldData	NVarChar(max)	Y	Host field data.

### Indexes

Name	Kind	Columns	Purpose
PKJnlHostDataFieldsUniqueID	P	JnlHostDataFieldID	Primary Key (non-clustered).
IXJnlHDFieldsJnlDraftID	IX	JnlDraftID	Used for loading all host fields associated with a payment.
IXJnlHDFieldsFieldID	IX	FieldID	Used for loading fields by type.

### <sup>1</sup> FieldID Values

Value	Gateway Constant Name	Description
100	HOST_DATA_FIELD_ID_BANK_USER_DATA	Bank user data typically included with a card token for additional identification.
101	HOST_DATA_FIELD_ID_EMV_TAG_4F	EMV Tag 4F - Application Identifier (AID)
102	HOST_DATA_FIELD_ID_EMV_TAG_50	EMV Tag 50 - Application Label
103	HOST_DATA_FIELD_ID_EMV_TAG_5F2A	EMV Tag 5F2A - Currency Code
104	HOST_DATA_FIELD_ID_EMV_TAG_5F34	EMV Tag 5F34 - Application PAN Sequence Number
105	HOST_DATA_FIELD_ID_EMV_TAG_82	EMV Tag 82 - Application Interchange Profile
106	HOST_DATA_FIELD_ID_EMV_TAG_84	EMV Tag 84 - Dedicated File Name
107	HOST_DATA_FIELD_ID_EMV_TAG_8A	EMV Tag 8A - Application Response Code (ARC)
108	HOST_DATA_FIELD_ID_EMV_TAG_95	EMV Tag 95 - Terminal Verification Results (TVR)
109	HOST_DATA_FIELD_ID_EMV_TAG_9A	EMV Tag 9A - Terminal Transaction Date
110	HOST_DATA_FIELD_ID_EMV_TAG_9B	EMV Tag 9B - Transaction Status Indicator (TSI)
111	HOST_DATA_FIELD_ID_EMV_TAG_9C	EMV Tag 9C - Cryptogram Transaction Type
112	HOST_DATA_FIELD_ID_EMV_TAG_9F02	EMV Tag 9F02 - Authorized Amount
113	HOST_DATA_FIELD_ID_EMV_TAG_9F03	EMV Tag 9F03 - EMV Cashback Amount
114	HOST_DATA_FIELD_ID_EMV_TAG_9F06	EMV Tag 9F06 - Terminal Application Identifier (AID)
115	HOST_DATA_FIELD_ID_EMV_TAG_9F07	EMV Tag 9F07 - Application Usage Control
116	HOST_DATA_FIELD_ID_EMV_TAG_9F08	EMV Tag 9F08 - ICC App Version Number
117	HOST_DATA_FIELD_ID_EMV_TAG_9F09	EMV Tag 9F09 - Term App Version Number
118	HOST_DATA_FIELD_ID_EMV_TAG_9F0D	EMV Tag 9F0D - Issuer Action Code - Default
119	HOST_DATA_FIELD_ID_EMV_TAG_9F0E	EMV Tag 9F0E - Issuer Action Code - Denial
120	HOST_DATA_FIELD_ID_EMV_TAG_9F0F	EMV Tag 9F0F - Issuer Action Code - Online
121	HOST_DATA_FIELD_ID_EMV_TAG_9F10	EMV Tag 9F10 - Issuer Application Data (IAD)
122	HOST_DATA_FIELD_ID_EMV_TAG_9F12	EMV Tag 9F12 - Application Preferred Name
123	HOST_DATA_FIELD_ID_EMV_TAG_9F1A	EMV Tag 9F1A - Terminal Country Code
124	HOST_DATA_FIELD_ID_EMV_TAG_9F1E	EMV Tag 9F1E - Interface Device (IFD) Serial Number
125	HOST_DATA_FIELD_ID_EMV_TAG_9F21	EMV Tag 9F21 - Transaction Time
126	HOST_DATA_FIELD_ID_EMV_TAG_9F26	EMV Tag 9F26 - Cryptogram
127	HOST_DATA_FIELD_ID_EMV_TAG_9F27	EMV Tag 9F27 - Cryptogram Info Data
128	HOST_DATA_FIELD_ID_EMV_TAG_9F33	EMV Tag 9F33 - Terminal Capabilities
129	HOST_DATA_FIELD_ID_EMV_TAG_9F34	EMV Tag 9F34 - CVM Result
130	HOST_DATA_FIELD_ID_EMV_TAG_9F35	EMV Tag 9F35 - Terminal Type
131	HOST_DATA_FIELD_ID_EMV_TAG_9F36	EMV Tag 9F36 - Application Transaction Counter
132	HOST_DATA_FIELD_ID_EMV_TAG_9F37	EMV Tag 9F37 - Unpredictable Number
133	HOST_DATA_FIELD_ID_EMV_TAG_9F39	EMV Tag 9F39 - POS Entry Mode
134	HOST_DATA_FIELD_ID_EMV_TAG_9F41	EMV Tag 9F41 - Transaction Sequence Counter
135	HOST_DATA_FIELD_ID_EMV_CHIP_INDICATOR	EMV Chip Indicator
136	HOST_DATA_FIELD_ID_EMV_CVM	EMV CVM (Cardholder Verification Method)
137	HOST_DATA_FIELD_ID_EMV_MODE	EMV Mode
138	HOST_DATA_FIELD_ID_TAC_DEFAULT	Terminal Action Code - Default
139	HOST_DATA_FIELD_ID_TAC_DENIAL	Terminal Action Code - Denial
140	HOST_DATA_FIELD_ID_TAC_ONLINE	Terminal Action Code - Online
141	HOST_DATA_FIELD_ID_ISSUER_SCRIPT_RESULTS	Issuer Script Results (EMV Tag 9F5B)
142	HOST_DATA_FIELD_ID_BUSINESS_ID	ScanPay - Business ID
143	HOST_DATA_FIELD_ID_TRANSACTION_ID	ScanPay - Transaction ID
144	HOST_DATA_FIELD_ID_EET_FIK	EET - Financial Code (FIK)
145	HOST_DATA_FIELD_ID_EET_BKP	EET - Security Code (BKP)
146	HOST_DATA_FIELD_ID_EET_PKP	EET - Signature Code (PKP)

147	HOST_DATA_FIELD_ID_SIHOST_ACCT	SIHOST.PMS - Account Number
148	HOST_DATA_FIELD_ID_KDS_ORDER_NUMBER	KDS (Kitchen Display) Order Number
149	HOST_DATA_FIELD_ID_KDS_OFFLINE	KDS (Kitchen Display) Offline
150	HOST_DATA_FIELD_ID_FIPAY_DEVICE_TRAN_ID	FIPay Device Transaction ID
151	HOST_DATA_FIELD_ID_EDC_TERMINAL_ID	EDC Terminal ID
152	HOST_DATA_FIELD_ID_EDC_MERCHANT_ID	EDC Merchant ID
153	HOST_DATA_FIELD_ID_EDC_RETRIEVAL_REF_NO	EDC Retrieval Reference Number
154	HOST_DATA_FIELD_ID_EDC_EXT_TRAN_ID	EDC External Transaction ID
155	HOST_DATA_FIELD_ID_EDC_EXT_TRAN_ID_PART2	EDC External Transaction ID (Continued)
156	HOST_DATA_FIELD_ID_EDC_BATCH_NO	EDC Batch Number
157	HOST_DATA_FIELD_ID_EDC_POS_REF_NO	EDC POS Reference Number
158	HOST_DATA_FIELD_ID_EDC_TRAN_DATETIME	EDC Host Transaction Date/Time
159	HOST_DATA_FIELD_ID_EDC_TENOR_VALUE	EDC Months of Tenor
160	HOST_DATA_FIELD_ID_EDC_MONTHLY_AMOUNT	EDC Monthly Amount
161	HOST_DATA_FIELD_ID_EDC_LAST_MONTH_AMOUNT	EDC Last Month Amount
162	HOST_DATA_FIELD_ID_EDC_CARD_TYPE	EDC Card Type
163	HOST_DATA_FIELD_CVM_PERFORMED_TEXT	Text representation of the CVM performed
164	HOST_DATA_FIELD_ID_PREFORMATTED_RECEIPT1	Receipt 1
165	HOST_DATA_FIELD_ID_PREFORMATTED_RECEIPT2	Receipt 2
166	HOST_DATA_FIELD_ID_PAGER_NUMBER	Pager Number
167	HOST_DATA_FIELD_ID_EMV_TAG_98	EMV Tag 98 - Transaction Certificate (TC) Hash Value
168	HOST_DATA_FIELD_ID_EDC_PIN_STATEMENT	EDC PIN Statement
169	HOST_DATA_FIELD_OCIUS25_CARDTYPE	Ocius25 Card Type
170	HOST_DATA_FIELD_OCIUS25_CAPTUREMETHOD	Ocius25 Capture Method
171	HOST_DATA_FIELD_OCIUS25_AUTHSERVER	Ocius25 Authentication Server
172	HOST_DATA_FIELD_OCIUS25_PTID	Ocius25 Permanent Terminal ID
173	HOST_DATA_FIELD_OCIUS25_SN	Ocius25 Terminal Serial Number
174	HOST_DATA_FIELD_OCIUS25_REFERENCE_NO	Ocius25 Reference Number
175	HOST_DATA_FIELD_ID_BUYER_ID	EDC Buyer ID
176	HOST_DATA_FIELD_ID_AUTH_AMOUNT	EDC Authorization Amount
177	HOST_DATA_FIELD_ID_ALIPAY_PAYMENT_ID	Alipay Payment ID
178	HOST_DATA_FIELD_ID_ALIPAY_PARTNER_TRANSACTION_ID	Alipay Partner Transaction ID
179	HOST_DATA_FIELD_ID_PRINT_SIGNATURE_LINE	Print signature line on receipt
180	HOST_DATA_FIELD_ID_RECEIPT_LANGUAGE_CODE	Language code (typically 2- or 3-digit ISO 639-1) for the language to use on the payment receipt

## 12.30 JnlItemQuantityReductions

The JnlItemQuantityReductions table provides supplementary information for the corresponding JnlDetails record when that detail refers to a item quantity reduction. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 1053.

### Columns

Column	Type	Allow Nulls	Description
JnlItemQuantityReductionID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
PLU	nvarchar(20)	Y	PLU of the item
Price	money	Y	Price of the reduction
Quantity	Int	Y	Quantity of the reduction

### Indexes

Name	Kind	Columns	Purpose
PKJnlItemQuantityReductionID	P	JnlItemQuantityReductionID	Primary Key.
CXJnlItemQuantityReductions		JnlTranID, JnlItemQuantityReductionID	Clustered Index. Used to load a list of records from the JnlItemQuantityReductions table where the JnlTranID falls within a specified range.

## 12.31 JnlItems

The **JnlItems** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to an item record. The **JnlCodeID** values of the corresponding record in the **JnlDetails** table include 102 (Items), 103 (Fees) and 104 (Donations).

### Columns

Column	Type	Allow Nulls	Description
JnlItemID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
Commission	Float	N	The commission earned for this ticket. The commission is based on the CommissionRate in the ticket's corresponding Chart of Accounts entry.
TranItemNo	Integer	N	Number of item in transaction.
Coupons	Integer	N	Number of coupons printed for this item.
Plu	Char(20)	N	This is the PLU of the item.
FkeyNo	Integer	N	FkeyNo is combination of the item's level and fkey numbers within the product: FkeyNo = (Level multiplied by 100) + Fkey.
ProductNo	Integer	N	Product from which the item was sold.
DiscNo	Integer	N	The ID number of the discount (defined in the Discount table) applied to this item, or 0 if none.
DiscAmt	Float	N	The monetary amount, in <i>base currency</i> , discounted from the item. Not used for price edits.
DiscOccur	Integer	N	A sequence number for the discount used in the transaction. For example, if three discounts are used in a transaction, one of the discounted tickets or items will have a DiscOccur value of 1, another of 2, and another of 3.
CostAmt	Float	N	Value of item "Cost" field, multiplied by the quantity. Although the "Cost" field of an item is intended as an optional way to store the cost of the item to the retailer (rather than to the customer), it is not actually used by the system for any purpose.
Tax	Float	N	This is the amount of a tax paid multiplied by the quantity. This includes tax charged upon additional payments made to increase the RemainingValue of the item.
TaxFlags	Integer	N	Bit mask for taxes. The bit is set if taxable for tax 1-8
TaxMethods	Char(8)	Y	The tax methods is an 8 character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket. <sup>1</sup>
TaxTableID	Integer	Y	If the item is taxed based on TaxTable, this links to the <b>TaxTableHeaders.TaxTableHeaderID</b>
SalesProgramID	Integer	Y	Foreign key to <b>SalesPrograms.SalesProgramID</b>
SupervisorID	Integer	Y	ID of user who gave supervisor approval
RedeemedValue	Float	Y	Points redeemed to purchase this item
LoyaltyPoints	Float	Y	The LoyaltyPoints column defines how many points were accrued for this item in a transaction.
UpsellStatus	Integer	Y	Indicates if there was no upsell journal record associated with this item record, or if this was an upsell selection. Also indicates if upsell choices were presented, but rejected. <sup>2</sup>
UpsellPLU	Varchar(20)	Y	The PLU from the upsell journal record associated with this item record. This is the original PLU that that was chosen that caused the user to select this item for upsell. This will be blank if this item was not added because of an upsell.
UpsellPriceDifference	Money	Y	The price difference from the upsell journal record associated with this item record. For replacement upsell, this is the difference in the price between the original PLU chosen, and the price of the upsell PLU. For add-on upsell, this is the full price of the upsell PLU. This is 0.00 if the item is not an upsell item.
UpsellType	Integer	Y	This is the upsell type from the upsell journal record associated with this item record. This is the type of upsell that determines how the upsell item was added to the transaction (as a replacement for the chosen item, or added with the original item). <sup>3</sup>
UpsellUserID	Integer	Y	The user ID of the user who performed the upsell. For POS transactions this will be the same as the transaction user. For order transactions, the upsell is performed by the user who adds the upsell items to the order, which is not necessarily the same as the user who issues the transaction.
UpsellSalesChannelType	Integer	Y	The sales channel type indicates where the upsell was performed (POS, OE, kiosk, Web). <sup>4</sup>
PkgInstanceDetailID	Integer	Y	Foreign key to <b>PkgInstanceDetails.PkgInstanceDetailID</b> . For returns of items in a package, this contains the ID of the item detail from the <b>PkgInstanceDetails</b> table.
PackagePrintSequence	Int	Y	The sequence number indicating the order in which the package detail was printed.
TransModifierID	Int	Y	Transaction-unique identifier for the modifier instance in the transaction that owns the item. This will match the <b>TransModifierID</b> on the modifier record in the journal that caused this item to be added.
GiftAidType	Integer	Y	Donation item's gift aid type. 0 indicates that there is no gift aid. <sup>5</sup>
UnitPrice	money	N	The unit price of an item.

### Indexes

Name	Kind	Columns	Purpose
PKJnlItemsJnlItemID	P	JnlItemID	Primary Key.
CXJnlItemsJnlTrnIDJnlItemID		JnlTrnID, JnlItemID	Clustered Index.  Used to load a list of records from the <b>JnlItems</b> table where the <b>JnlTrnID</b> falls within a specified range.

<sup>1</sup> TaxMethod values, per character

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis.
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis.
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary.
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis.

<sup>2</sup> UpsellStatus Values

Value	Gateway Constant Name	Description
0	UPSELL_STATUS_NO_UPSELL	No upsell record exists for this item
1	UPSELL_STATUS_SELECTED	This item was created as the result of an upsell
2	UPSELL_STATUS_CANCELLED	Upsell options were presented for this item, but were rejected.

3	UPSELL_STATUS_TRANSACTIONAL	This item was sold as part of a transactional upsell.
---	-----------------------------	---

**3 UpsellType Values**

Value	Gateway Constant Name	Description
0	UPSELL_TYPE_REPLACEMENT	This item was added to replace the original PLU selected
1	UPSELL_TYPE_ADD_ON	This item was added in addition to the original PLU selected

**4 UpsellSalesChannelType Values**

Value	Gateway Constant Name	Description
0	sctPOS	The upsell was performed from the Point of Sale
1	sctOrderEntry	The upsell was performed from Order Entry
2	sctKiosk	The upsell was performed from the Kiosk
3	sctWebStore	The upsell was performed from the WebStore

**5 GiftAidType Values**

Value	Gateway Constant Name	Description
0	gaNone	No GiftAid was sold with the item.
2	gaFull	GiftAid is for the full price of the item. This is currently only allowed for donation items.

## 12.32 JnlItineraries

Stores the Segment records from the journal.

### Columns

Column	Type	Allow Nulls	Description
JnlItineraryID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Details
LegNo	Int	Y	Segment number
Origin	Int	Y	Origin code
Destination	Int	Y	Destination code
Miles	Int	Y	Number of Miles to Destination
LegValue	Money	Y	Value of Segment
Tariff	VarChar(20)	Y	Tariff code
CompanyNo	Int	Y	Carrier code
Carrier	Char(4)	Y	Carrier Abbreviation
Schedule	Char(4)	Y	Schedule code
JnlTripSummaryID	Int	Y	Link back to Jnl Trip Summary table
Direction	Int	Y	Indicates the portion of the itinerary for this leg. "0" for Outbound, "1" for the Inbound portion of the trip
Seg1	Int	Y	From TTrip.SelectedItinerary.Legs[i].Seg1 and Seg1. Seg1 is the origin segment on "Schedule". Seg1 and Seg2 point to the first and last segment of the schedule that the passenger is actually riding. For example, the schedule may run ABCDEF but this ticket is only for BCD on the schedule. In that case, Seg1 will be 2 and Seg2 will be 4.
Seg2	Int	Y	The destination segment on "Schedule".

### Indexes

Name	Kind	Columns	Purpose
PKJnlItineraryID	P	JnlItineraryID	Primary Key.

### 12.33 JnlLoyaltyAccounts

The JnlLoyaltyAccounts Table contains loyalty account information collected during a loyalty transaction.

#### Columns

Column	Type	Allow Nulls	Description
JnlLoyaltyAccountID	Integer	N	Primary key, always unique
JnlTranID	Integer	N	Associated journal transaction
LoyaltyProgramID	Integer	N	Loyalty program associated with the loyalty account
AccountNo	Varchar(40)	N	Loyalty Account number
PointExpirationDate	DateTime	Y	The date points expire for the loyalty accrual transaction. If the column is null, the points never expire.
PluginPointsAccrued	Float	Y	Number of points accrued through a loyalty plugin for this transaction. Will be 0 if the loyalty account is not from a loyalty plugin. Used for population of keywords on reprint receipts.
PluginBalance	Float	Y	Loyalty plugin account balance after points accrual. Will be 0 if the loyalty account is not from a loyalty plugin. Used for population of keywords on reprint receipts.
ExternalTransID	NVarChar(20)	Y	External transaction ID received from the plugin. This is used for plugins where the external system requires its own transaction ID to be used for voiding a transaction.

#### Indexes

Name	Kind	Columns	Purpose
PKJnlLoyaltyAccountID	P	JnlLoyaltyAccountID	Primary Key.

## 12.34 JnlModifiers

The JnlModifiers table stores food and beverage item modifier information.

### Columns

Column	Type	Allow Nulls	Description
JnlModifierID	Int	N	Primary key, always unique
JnlItemID	Int	N	Foreign key to JnlItems.JnlItemID. Indicates the item that was modified.
ModifierID	Int	N	Foreign key to Modifiers.ModifierID. Indicates the modifier that was selected for the item.
MemoID	Int	Y	Foreign key to MemoTbl.MemoTblID. For special instruction modifiers, indicates the memo containing the special instruction text.
TransModifierID	Int	Y	Unique identifier of the modifier instance in the transaction. Used to link a modifier to its associated items.
ParentTransModifierID	Int	Y	Unique identifier of the modifier instance in the transaction that this modifier was nested under.

### Indexes

Name	Kind	Columns	Purpose
PKJnlModifiersUniqueID	P	JnlModifierID	Primary key
IXJnlModifiersJnlItemID	IX	JnlItemID	Used to load modifiers for a specific journal item.
IXJnlModifiersModifierID	IX	ModifierID	Used to load modifiers of a specific type.

## 12.35 JnlNoSales

The JnlNoSales table provides supplementary information for the corresponding JnlDetails record when that detail refers to a no sale transaction.

### Columns

Column	Type	Allow Nulls	Description
JnlNoSaleID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
DrawerNo	Int	N	Number of the cash drawer opened by the no sale online function

### Indexes

Name	Kind	Columns	Purpose
PKJnlNoSalesJnlNoSaleID	P	JnlNoSaleID	Primary Key.
CXJnlNoSalesJnlTranIDJnlNSID		JnlTranID, JnlNoSaleID	Clustered Index. Used to load a list of records from the JnlNoSales table where the JnlTranID falls within a specified range.

## 12.36 JnlOverShorts

This is an auxiliary journal table and will store the values from the OVER\_SHORT journal record. Part of the OVER\_SHORT record information was already stored in the JnlDetails table. But a new field has been added to this record and this new field will be stored in the JnlOverShorts table that complements JnlDetails table for OVER\_SHORT records.

### Indexes and Constraints

Primary Key: JnlItemID

Indexes:

(None)

Column	Type	Allow Nulls	Description
JnlOverShortID	Int	N	Unique. Primary Key. Generated from GatewayCounters.
JnlTranID	Int	N	References JnlHeader.JnlTranID.
Status	Int	N	Cashout status. See the list of values below. <sup>1</sup>
ShiftStartTime	DateTime	Y	Date and time captured at beginning of new shift
ShiftStartTransNo	Integer	Y	Online Transaction Number captured at beginning of new shift

### Indexes

Name	Kind	Columns	Purpose
PKJnlOverShortJnlOverShortID	P	JnlOverShortID	Primary Key.
CXJnlOverShortsJnlTrnIDJnlOSID		JnlTranID, JnlOverShortID	Clustered Index. Used to load a list of records from the JnlOverShorts table where the JnlTranID falls within a specified range.

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	csLocal	Local Cashout
1	csRemoteComplete	Remote cashout with all necessary data available at SQL database at the cashout time.
2	csRemoteLastTransactionError	Remote cashout: the last transaction of the current node was not at SQL database at the cashout time. This means that the current shift has not been completely DBSynched at that time.
3	csRemoteShiftsNotSuspended	Remote Cashout: Some of the shifts been cashed out are still opened either because the cast member did not suspended the shift or because the SUSPEND record has not been DBSynched yet.
4	csRemoteIncompleteShiftList	None of the transactions of at least one shift that should take part on the cashout were DBSynched. Then the SQL Database still does not know about these shifts. The sales amounts of these shift will not be taking in consideration when calculating the over short amount.

## 12.37 JnlPackages

This table stored header information that corresponds to package sales. Each JnlPackage row will have one to many corresponding details. These details can be found in the SuperTickets table by joining on the VisualID field.

### Columns

Column	Type	Allow Nulls	Description
JnlPackageID	Int	N	Primary key, always unique
JnlTranID	Int	Y	Value that links to JnlHeaders.JnlTranID. Can be used to collect records for an entire transaction.
PLU	Varchar(20)	Y	The PLU of the package. Links to Items.PLU
VisualID	Varchar(40)	Y	The ID of the package. Links to SuperTickets.VisualID
UpsellStatus	Integer	Y	Indicates if there was no upsell journal record associated with this package record, or if this was an upsell selection. Also indicates if upsell choices were presented, but rejected. <sup>1</sup>
UpsellPLU	Varchar(20)	Y	The PLU from the upsell journal record associated with this package record. This is the original PLU that was chosen that caused the user to select this package for upsell. This will be blank if this package was not added because of an upsell.
UpsellPriceDifference	Money	Y	The price difference from the upsell journal record associated with this package record. For replacement upsell, this is the difference in the price between the original PLU chosen, and the price of the upsell PLU. For add-on upsell, this is the full price of the upsell PLU. This is 0.00 if the package is not an upsell item.
UpsellType	Integer	Y	This is the upsell type from the upsell journal record associated with this package record. This is the type of upsell that determines how the upsell item was added to the transaction (as a replacement for the chosen item, or added with the original item). <sup>2</sup>
UpsellUserID	Integer	Y	The user ID of the user who performed the upsell. For POS transactions this will be the same as the transaction user. For order transactions, the upsell is performed by the user who adds the upsell items to the order, which is not necessarily the same as the user who issues the transaction.
UpsellSalesChannelType	Integer	Y	The sales channel type indicates where the upsell was performed (POS, OE, kiosk, Web). <sup>3</sup>
LoyaltyPoints	Float	Y	LoyaltyPoints stores the number of loyalty points earned when selling a Package with the pricing method configured to obtain the price from the package.
GiftAidAmount	Money	Y	The gift aid amount for the package
GiftAidType	Integer	Y	The type of gift aid sold with the package. 0 indicates that there is no gift aid. <sup>4</sup>
PassID	Int	Y	The unique ID of the pass this package is associated with. Links to Passes.PassNo.
Reprinted	Bit	Y	Indicates that the product has already been reprinted.

### Indexes

Name	Kind	Columns	Purpose
PKJnlPackagesJnlPackageID	P	JnlPackageID	Primary Key.
CXJnlPackagesJnlTranIDJnlPkgID		JnlTranID, JnlPackageID	Clustered Index. Used to load a list of records from the JnlPackages table where the JnlTranID falls within a specified range.
IXJnlPackagesVisualID		VisualID	Speed query performance

#### <sup>1</sup> UpsellStatus Values

Value	Gateway Constant Name	Description
0	UPSELL_STATUS_NO_UPSELL	No upsell record exists for this package
1	UPSELL_STATUS_SELECTED	This package was created as the result of an upsell
2	UPSELL_STATUS_CANCELLED	Upsell options were presented for this package, but were rejected.
3	UPSELL_STATUS_TRANSACTIONAL	This item was sold as part of a transactional upsell.

#### <sup>2</sup> UpsellType Values

Value	Gateway Constant Name	Description
0	UPSELL_TYPE_REPLACEMENT	This package was added to replace the original PLU selected
1	UPSELL_TYPE_ADD_ON	This package was added in addition to the original PLU selected

#### <sup>3</sup> UpsellSalesChannelType Values

Value	Gateway Constant Name	Description
0	sctPOS	The upsell was performed from the Point of Sale
1	sctOrderEntry	The upsell was performed from Order Entry
2	sctKiosk	The upsell was performed from the Kiosk
3	sctWebStore	The upsell was performed from the WebStore

#### <sup>4</sup> GiftAidType Values

Value	Gateway Constant Name	Description
0	gaNone	No gift aid is associated
1	gaPercentage	Percentage-based gift aid
2	gaFull	Gift aid was used for the full amount of the package

## 12.38 JnlPaidIOs

The **JnlPaidIOs** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a paid in or paid out record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 210 (paid in) or 310 (paid out) or 420 (tax refund).

### Columns

Column	Type	Allow Nulls	Description
JnlPaidOID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
Descr	Char(24)	N	Description.
CommissionAmount	Float	N	The commission earned for this ticket. The commission is based on the CommissionRate in the ticket's corresponding Chart of Accounts entry.
ProductID	Integer	N	Product from which the "ticket" was issued.
Fkey	Integer	N	FkeyNo is combination of the "ticket" type's level and fkey numbers within the product: FkeyNo = (Level multiplied by 100) + Fkey.
PLU	Char(20)	Y	The <b>JnlPaidOs.PLU</b> column represents the PLU used for the paid in/out. This data is populated in the local journal in the <b>EXTRA_PAID_IO_REC</b> record.
SupervisorID	Int	Y	ID of the Supervisor that approved this Paid In/Out. FK reference to <b>GxUsers.UserID</b>

### Indexes

Name	Kind	Columns	Purpose
PKJnlPaidIOSJnlPaidOID	P	JnlPaidOID	Primary Key.
CXJnlPaidIOSJnlTrnIDJnlPdOID		JnlTranID, JnlPaidOID	Clustered Index. Used to load a list of records from the <b>JnlPaidIOS</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.39 JnlPayments

The **JnlPayments** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a payment record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 532.

### Columns

Column	Type	Allow Nulls	Description
JnlPaymentID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
Fop	Integer	Y	Foreign key to <b>FOPs.FOPID</b> , specifying the form of payment used for this payment.
ExchAmount	Float	Y	Payment amount represented in the type of currency used.
Descr	VarChar(50)	Y	This description field usually contains endorsement information (if any) entered for the form of payment (such as a credit card number). It can also contain a customer number when the system journalizes a charge account or invoice payment. If the form of payment is configured to allow cashback, "Cashback" is used for cashback payments. For tender exchange ("TENDEX") payments, this field contains the text entered into the "Document" field for this payment.
GxKeyID	Int	Y	The ID that corresponds to the Key which will decode the encoded Credit Card number, saved in the <b>Descr</b> column.
PreAuthCode	Char(10)	Y	For returns, this field contains the approval code of the return authorization (not the approval code entered for the original sale).
MPOriginalJnlTranID	Int	Y	The <b>JnlTranID</b> of the original journal transaction where the tickets were sold. This is populated in a multi-payment return transaction from the Point of Sale.

### Indexes

Name	Kind	Columns	Purpose
PKJnlPaymentsJnlPaymentID	P	JnlPaymentID	Primary Key.
CXJnlPaymentsJnlTrnIDJnlPmntID		JnlTranID, JnlPaymentID	Clustered Index.  Used to load a list of records from the <b>JnlPayments</b> table where the <b>JnlTranID</b> falls within a specified range.
IXJnlPaymentsJTIDJPIDFop	AK	JnlTranID, JnlPaymentID, Fop	Optional: Used to speed up query for loading prepayment history information when viewing prepayment history from the Customer Payments screen, and from a prepayment report. To add this index run the GalaxyDatabase script with @ AddIndexesForPrePayment set to 1.
IXJnlPaymentsMPOrigJTID	AK	MPOriginalJnlTranID	Improve load performance for returns.

## 12.40 JNLPTOs

### Columns

Column	Type	Allow Nulls	Description
JnlPTOID	Int	N	Unique ID
JnlTranID	Int	N	Link to JLDetails
FullFareQty	Int	N	Number of full fare passengers
HalfFareQty	Int	N	Number of Half fare passengers
TicketKind	Int	N	One way or round trip
Origin	Int	N	Trip starting point
Destination	Int	N	Trip ending point
Fare	Money	N	Price of trip
Fee	Money	N	Fee charged to sell PTO
Advance	Money	N	Cash advance
AdvanceFee	Money	Y	Fee charged for cash advance

### Indexes

Name	Kind	Columns	Purpose
PKJnlPTOID	P	JnlPTOID	Primary Key.

## 12.41 JnlReasons

The **JnlReasons** table contains data recorded in JNL.DAT for reasons. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 54.

### Columns

Column	Type	Allow Nulls	Description
JnlReasonID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID.
ReasonID	Integer	N	Reason ID used during the transaction.
ReasonType	Integer	Y	Indicates the reason type from the selected reason at the time of journalization <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKJnlReasonsJnlReasonID	P	JnlReasonID	Primary Key.
CXJnlReasonsJnlTranIDJnlReasonID		JnlTranID, JnlReasonID	Clustered Index. Used to load a list of records from the JnlReasons table where the JnlTranID falls within a specified range.

<sup>1</sup> ReasonType Values

Gateway Constant Name	Value	Description
rtTicketReason	0	Ticket sale reason
rtPaymentRefundReason	1	Reason for payment refund

## 12.42 JnlReceipts

The **JnlReceipts** table records when Summary or Detail receipts were printed, if the system is configured to journalize them (in the Online Configuration). The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 52.

### Columns

Column	Type	Allow Nulls	Description
JnlReceiptID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to the <b>JnlHeaders.JnlTranID</b>
ReceiptNo	Integer	N	System receipt number for Summary receipts, 0 for Detail receipts
ReceiptKind	Char(1)	N	D for Detail receipts, or S for Summary receipts
OrigTransNo	Integer	N	Transaction number ( <b>JnlHeader.TranNo</b> ) of the reprinted transaction
TransAmount	Float	N	Total purchase/return amount, including tax
ReceiptQty	Integer	N	Always has value 1
ReprintCount	Integer	Y	Number of times the receipt has been reprinted.

### Indexes

Name	Kind	Columns	Purpose
PKJnlReceiptsJnlReceiptID	P	JnlReceiptID	Primary Key.
CXJnlReceiptsJnlTranIDJnlReceiptID		JnlTranID, JnlReceiptID	Clustered Index. Used to load a list of records from the <b>JnlReceipts</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.43 JnlRecharges

The **JnlRecharges** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a recharge record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 62. A transaction will consist of one recharge record for each ticket record that was a debit card recharge.

### Columns

Column	Type	Allow Nulls	Description
JnlRechargeID	Int	N	Primary key, always unique.
JnlTranID	Int	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
VisualID	VarChar(60)	Y	Foreign key to <b>DebitCards.VisualID</b> .
GxKeyID	Int	N	Foreign key to <b>GxKeys.GxKeyID</b> , references encryption key used to encrypt the debit card number ( <b>VisualID</b> ).
DataConnection	Int	Y	Indicates the stored value module used for this given recharge
HostData	Varchar(255)	Y	Used to hold data journalized by the host system that may be needed in a void of a Debit/Draft/Recharge.
DebitTypeID	Int	Y	Foreign key to <b>DebitTypes.DebitTypeID</b> . Indicates the debit type that was used for the recharge.
ProductID	Int	Y	ID of the product that was added to the SV card.
ProductPrice	Money	Y	Price of the product that was added to the SV card.

### Indexes

Name	Kind	Columns	Purpose
PKJnlRechargesJnlRechargeID	P	JnlRechargeID	Primary Key.
CXJnlRechrgsJnlTrnIDJnlRchrgID		JnlTranID, JnlRechargeID	Clustered Index.  Used to load a list of records from the <b>JnlRecharges</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.44 JnlReissues

This table contains the reissuance information for a ticket.

### Columns

Column	Type	Allow Nulls	Description
JnlReissueID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Detail
Company	Int	N	COA information
Category	Int	N	COA information
SubCat	Int	N	COA information
Amount	Money	N	Price of Reissued ticket
Serial	VarChar(40)	N	Serial number of old ticket
Tax1	Money	Y	Amount of tax 1
Tax2	Money	Y	Amount of tax 2
Tax3	Money	Y	Amount of tax 3
Tax4	Money	Y	Amount of tax 4
Tax5	Money	Y	Amount of tax 5
Tax6	Money	Y	Amount of tax 6
Tax7	Money	Y	Amount of tax 7
Tax8	Money	Y	Amount of tax 8

### Indexes

Name	Kind	Columns	Purpose
PKJnlReissueID	P	JnlReissueID	Primary Key.

## 12.45 JnlRenewalCredits

This table holds the visual ID of a ticket applied to a pass renewal as a renewal credit.

### Columns

Column	Type	Allow Nulls	Description
JnlRenewalCreditID	Int	N	Primary key, always unique
JnlTranID	Int	N	Journal transaction ID
VisualID	Varchar (40)	N	Previous Visual ID

### Indexes

Name	Kind	Columns	Purpose
PKJnlRenewalCreditID	P	JnlRenewalCreditID	Primary Key.
IXJnlRenewalCreditsVisualID	A	VisualID	Alternate key.
IXJnlRenewalCreditsJnlTranID	A	JnlTranID	Alternate Key.

## 12.46 JnlReprints

This table holds the previous visual ID of a reprinted ticket.

### Columns

Column	Type	Allow Nulls	Description
JnlReprintID	Int	N	Primary key, always unique
JnlTranID	Int	N	Journal transaction ID
VisualID	Varchar (40)	N	Previous Visual ID
NewVisualID	Varchar(40)	Y	Visual ID of the new reprinted ticket

### Indexes

Name	Kind	Columns	Purpose
PKJnlReprintsJnlReprintID	P	JnlReprintID	Primary Key.
CXJnlReprntsJnlTrnIDJnlRpntID		JnlTranID, JnlReprintID	Clustered Index. Used to load a list of records from the JnlReprints table where the JnlTranID falls within a specified range.

## 12.47 JnlResellerDetails

The **JnlResellerDetails** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a reseller detail record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 51. The **JnlResellerDetails** table provides additional price information about tickets that were sold through the reseller web store. This information includes the price that the ticket was sold at by the reseller, and the price that the ticket was sold to the reseller by the Galaxy user. These prices are originally obtained from the PLU detail record from the sales channel category.

Columns

Column	Type	Allow Nulls	Description
JnlResellerDetailID	Int	N	Primary key, always unique
JnlTranID	Int	N	Journal transaction ID
VisualID	Varchar(40)	N	Visual ID of the ticket that the prices are associated with
ExternalPrice	Money	N	Price at which the ticket was sold to the reseller.
RetailPrice	Money	N	Price at which the ticket was sold to the customer by the reseller.
ExternalPLU	NVarChar(100)	Y	The item number set by the local administrator (Reseller PLU).
BookingReference	NVarChar(40)	Y	Booking information provided by the reseller.
TravelDate	DateTime	Y	Expected date of travel provided by the reseller. This date does not factor in to any pricing decisions, it is strictly informational.

Indexes

Name	Kind	Columns	Purpose
PKJnlRsllrDtlsJnlRsllrDtIID	P	JnlResellerDetailID	Primary key
CXJnlRsllrDtlsJnlRsllrDtIID		JnlTranID, JnlResellerDetailID	Clustered Index  Used to load a list of records from the JnlReservations table where the JnlTranID falls within a specified range.
IXJnlRsllrDtlsJnlTranIDVisualID	A	JnlTranID, VisualID	Alternate key. Individual tickets will be referenced by JnlTranID and VisualID for reports
IXJnlRsllrDtlsVisualID	A	VisualID	Alternate key. Used to look up sale and return prices for a specific ticket

## 12.48 JnlReservations

The **JnlReservations** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a reservation. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 48.

### Columns

Column	Type	Allow Nulls	Description
JnlReservationID	Int	N	Primary key, always unique.
JnlTranID	Int	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
ReservationID	Int	Y	ReservationID selected by TS; Foreign key to <b>RMReservations.ReservationID</b> .
ResourceID	Int	Y	ResourceID associated with ReservationID; Foreign key to <b>RMResources.ResourceID</b> .
ResourceTypeID	Int	Y	ResourceTypeID for the Resource; Foreign key to <b>RMResourceTypes.ResourceTypeID</b> .
StartDate	DateTime	Y	Start date and time of the reservation
EndDate	DateTime	Y	End date and time of the reservation.

### Indexes

Name	Kind	Columns	Purpose
PKJnlReservationsJnlResID	P	JnlReservationID	Primary Key.
CXJnlRsrvtnsJnlTrnIDJnlRsvtnID		JnlTranID, JnlReservationID	Clustered Index  Used to load a list of records from the <b>JnlReservations</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.49 JnlStocks

The **JnlStocks** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a ticket stock record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is in the range 28 to 32.

### Columns

Column	Type	Allow Nulls	Description
JnlStockID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
StockNodeNo	Integer	N	Node number.
ShiftNo	Integer	N	Shift number.
StockNo	Integer	N	Stock number defined in ticket type configuration.
ProductNo	Integer	N	Product number, or 0.
Fkeyld	Integer	N	Level number multiplied by 100 added to function key number, or 0.
Descr	Char(24)	N	Name of ticket stock.
Coupons	Integer	N	Total number of coupons.
SerialLow	Integer	N	Always 0. This column is not currently used by the system.
SerialHigh	Integer	N	Always 0. This column is not currently used by the system.

### Indexes

Name	Kind	Columns	Purpose
PKJnlStocksJnlStockID	P	JnlStockID	Primary Key.
CXJnlStocksJnlTranIDJnlStockID		JnlTranID, JnlStockID	Clustered Index. Used to load a list of records from the <b>JnlStocks</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.50 JnlSuperTickets

### Columns

Column	Type	Allow Nulls	Description
JnlSuperTicketID	Integer	N	Primary key, always unique.
VisualID	Varchar(40)	Y	
ContactID	Integer	Y	References CustContacts.CustContactID
UseReplenish	Bit	Y	Flag to denote if the Replenish Status field should be used
ReplenishStatus <sup>1</sup>	Integer	Y	Status of the Replenishment
JnlTranID	Integer	Y	Links super ticket records to transaction header
SuperTicketType	Int	Y	<sup>2</sup> Identifies the type of super ticket record referenced by the journal record.

### Indexes

Name	Kind	Columns	Purpose
PKJnlSuperTicketID	P	JnlSuperTicketID	Primary Key.

### <sup>1</sup> ReplenishStatus Values

Value	Gateway Constant Name	Description
0	ST_NOT_REPLENISH_STATUS	Can not be replenished
1	ST_ACTIVE_REPLENISH_STATUS	Replenish Active
2	ST_REPLENISHED_REPLENISH_STATUS	Replenish has been used
3	ST_CANCEL_REPLENISH_STATUS	Replenish has been cancelled
4	ST_SUSPEND_REPLENISH_STATUS	Replenish has been suspended

### <sup>2</sup> SuperTicketType Values

Value	Gateway Constant Name	Description
0	JNL_ST_TYPE_STANDARD	Standard super ticket journal record
1	JNL_ST_TYPE_JOINT_MEMBER	Joint member super ticket journal record

## 12.51 JnlSurcharges

A surcharge is the additional monies charged with a fare during peak times (i.e. Holidays, weekends, etc.).

### Columns

Column	Type	Allow Nulls	Description
JnlSurchargeID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link back to main transaction
Name	Varchar(24)	N	Name of Surcharge
Amount	money	N	Money of surcharge
Basis <sup>1</sup>	char	N	How surcharge is applied

### Indexes

Name	Kind	Columns	Purpose
PKJnlSurchargeID	P	JnlSurchargeID	Primary Key.
CXJnlSurcharge	C	JnlTranID, JnlSurchargeID	Clustered Index

### <sup>1</sup> Basis Values

Value	Description
F	Lookup is by Fare
M	Lookup is by miles

## 12.52 JnlSurveys

The **JnlSurveys** table contains survey answers recorded in the sales journal. The JnlCodeID value of the corresponding **JnlDetails** table is 55.

### Columns

Column	Type	Allow Nulls	Description
JnlSurveyID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID.
SurveyID	Integer	N	ID of the survey that was performed.
AnswerNumber	Integer	N	Question number.
AnswerText	Char(40)	N	Answer given in response to the survey question.
SurveyFieldID	Integer	Y	Foreign key reference to the SurveyFields.SurveyFieldID to help link the Answer to its question.

### Indexes

Name	Kind	Columns	Purpose
PKJnlSurveysJnlSurveyID	P	JnlStockID	Primary Key.
CXJnlSurveysJnlTrnIDJnlSurveyID		JnlTranID, JnlSurveyID	Clustered Index.  Used to load a list of records from the JnlSurveys table where the JnlTranID falls within a specified range.

## 12.53 JnlSVLookups

This table contains stored value lookup records. If configured to do so, the POS will record a journal transaction each time a stored value card is looked up.

### Columns

Column	Type	Allow Nulls	Description
JnlSVLookupID	Integer	N	Primary key, always unique.
JnlTranID	Int	Y	Foreign key to JnlHeaders.JnlTranID.
CardNumber	Varchar(60)	Y	The card number that was looked up
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, references encryption key used to encrypt the stored value card (CardNumber).

### Indexes

Name	Kind	Columns	Purpose
PKJnlSVLookupsJnlSVLookupID	P	JnlSVLookupID	Primary key.
CXJnlSVLkupsJnlTrnIDJnlSVLkpID		JnlTranID, JnlSVLookupID	Clustered Index.  Used to load a list of records from the JnlSVLookups table where the JnlTranID falls within a specified range.

## 12.54 JnlTaxes

The JnlTaxes table provides supplementary information for the corresponding JnlDetails record when that detail refers to a tax record. The JnlCodeID value of the corresponding record in the JnlDetails table is 120.

### Columns

Column	Type	Allow Nulls	Description
JnlTaxesID	Int	N	Primary key, always unique.
JnlTranID	Int	N	Foreign key reference to JnlHeaders.JnlTranID
Category	int	N	Number of the tax for this detail. (1 - 8)
Name	nvarchar(8)	N	Name of the tax.
Amount	money	N	Total amount of taxes collected for this tax category.
Rate	float	N	Tax rate for this tax category.

### Indexes

Name	Kind	Columns	Purpose
PKJnlTaxesID	P	JnlTaxesID	Primary Key.
CXJnlTaxes		JnlTranID, JnlTaxesID	Used to load a list of records from the JnlTaxes table where the JnlTranID falls within a specified range.

## 12.55 JnlTickets

The **JnlTickets** table contains ticket records from the journal. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 101.

### Columns

Column	Type	Allow Nulls	Description
JnlDetailID	Integer	N	Primary key, always unique.
VisualID	Char(40)	N	Barcode as read from the scanner.
AccessCode	Integer	N	Foreign key to AccessCodes.AccessCode, corresponds an access code definition specifying ticket type information with respect to Access Control in general (as opposed to ticket type information found in Products). The AccessCode of a ticket also affects how the ticket is used in most Access Control reports.
ID	Char(20)	N	A unique string identifying the ticket within the set of all tickets with the same AccessCode. The ID consists of the NodeNo and Serial values (below) zero padded and concatenated. The lengths are user definable, but the defaults are 3-digit node and 6-digit serial. Since a node may have multiple sets of serial numbers for multiple companies, each company should be assigned its own set of mutually exclusive AccessCodes when the system's database is developed.
NodeNo	Integer	N	Node number of the computer which issued the ticket.
TransNo	Integer	N	The sequential ID number of the transaction from which the ticket belongs. NodeNo and TransNo may be used together to look up transaction data in the TransHdr table.
LineNum	Integer	N	LineNum is a sequential journal detail count. If LineNum=8 then the JnlTicket is the 8th journal detail for the transaction.
Serial	Integer	N	The ticket's serial number for the node and company which issued the ticket.
Company	Integer	N	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
TktCode	Char(5)	N	The category and sub-category from the Chart Of Accounts.
Qty	Integer	N	The number of guests to be admitted with this ticket. Regular tickets always have a quantity of 1. Group tickets will have some number of guests which was selected during the sale.
TktIndex	Integer	N	The position of this ticket within the transaction for a specific ticket type. Each ticket in a transaction is uniquely identified by combining ProductNo, FkeyNo, and TktIndex. Each ticket record of a disbursed ticket will have the same TktIndex value. These may be uniquely identified by including the serial number, if used.
Coupons	Integer	N	Number of coupons printed for this ticket.
ProductNo	Integer	N	Product from which the ticket was issued.
FkeyNo	Integer	N	FkeyNo is combination of the ticket type's level and fkey numbers within the product: FkeyNo = (Level * 100) + Fkey.
DiscNo	Integer	N	Foreign key to Discounts.DiscountID, specifying the discount applied to this ticket, or 0 for no discount.
DiscAmt	Float	N	The amounts, in <i>base currency</i> , discounted from the ticket. If DiscNo is 0 and this column is used, this indicates that the ticket price was edited at the time of sale. In this case, DiscAmt contains the difference between the base price and the price used.
DisclIndex	Integer	N	A sequence number for the discount used in the transaction. For example, if three discounts are used in a transaction, one of the discounted tickets or items will have a DisclIndex value of 1, another of 2, and another of 3.
DateSold	Datetime	N	Date and time at which the ticket was issued.
Status	Integer	N	A number indicating the state of the ticket or pass as defined by the tables <i>Ticket Status Values</i> <sup>1</sup> and <i>Pass Status Values</i> <sup>2</sup> below. A ticket or pass may also be expired, but because that state may change without intervention by the system, it must be determined, as needed, by the system.
RemainingValue	Float	N	A floating point numeric value representing the value of the ticket. Typically, this is a number of admissions or a monetary value, but may represent any unit desired. For a regular ticket, the value would be one, or some number of guests or admissions. In a debit-card implementation, the number would represent the amount of money remaining on the ticket or card.
LastUse	Datetime	Y	The date and time that the ticket was last used by a guest. More specifically, the last time the record in the Tickets table was updated after being scanned at a scanning station.
LastACP	Integer	Y	Foreign key to ACPS.AcpID, specifying the Access Control Point number of the scanning station at which the ticket was last scanned.
UseCount	Integer	Y	Number of times the ticket has been used for admission or redemption of value.
Expiration	Datetime	Y	If not null, an expiration date which overrides any expiration information specified in the ticket's Access record (referenced by AccessCode).
CustNo	Char(10)	N	This is the account number of the customer who purchased this ticket. This column is blank if the ticket was not purchased by a customer with an account in the system, or if the ticket was added <i>dynamically</i> at the time of its first scan.
Orderno	Integer	N	Foreign key to Orders.OrderNo, specifying the order that the ticket was placed under (or 0 for no order).
EventNo	Integer	N	Foreign key to RMEvents.EventID, specifying the event that the ticket was sold from (or 0 for no event).
Price	Money	N	Price paid for ticket, excluding taxes and additional payments made to increase the RemainingValue of the ticket.
Tax	Money	N	Total amount of tax paid with ticket. This includes tax charged upon additional payments made to increase the RemainingValue of the ticket.
Taxes	Char(8)	N	An 8 character string containing an array of Y/N flags indicating which of the 8 possible taxes are included in the Tax value.
Commission	Money	N	The commission earned for this ticket. The commission is based on the CommissionRate in the ticket's corresponding Chart of Accounts entry.
Preprinted	Bit	N	This flag is set for pre-printed ticket types. It indicates that the serial number (in both the ID and Serial columns) is based on a user-defined value, not the company's current sequential serial number.
UseQty	Integer	N	The number of times the ticket has been used.
Act	Integer	N	The Chart of Accounts number. Defined as (Company * 100000) + (Category * 100) + Sub-Category
Duplicate	Integer	N	Not Currently Used
TicketDate	Datetime	Y	Used for Date Specific Tickets
PLU	Char(20)	Y	Foreign key to Items.PLU, this is the PLU for the ticket. This can either be in the form TICKETPPPLLFF for tickets associated with a product, or user-defined for tickets not associated with a product.
TaxMethods	Char(8)	Y	The tax methods is an 8 character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket. <sup>3</sup>
UpdateCode	Int	Y	Specifies whether or not sales data for this ticket was received from the POS. <sup>4</sup>
CustomerID	Int	Y	Populated with the internal Customer ID value when a tickets is sold in a customer-based transaction from the POS or OE. Foreign key reference to Customers.CustomerID
TaxTableID	Int	Y	Holds ID of a tax table, if one was used. Foreign Key reference to TaxTableHeader.TaxTableHeaderID
SalesProgramID	Integer	Y	Foreign key to SalesPrograms.SalesProgramID
ResourceID	Integer	Y	Foreign key to RMResources.ResourceID. If this is an event ticket, this column points to the resource that the event ticket was sold for.
Points	Integer	Y	The number of points sold on a points based debit card.
CapacityID	Integer	Y	The unique ID of the related RMCapacity record
SupervisorID	Integer	Y	ID of the user who gave supervisor approval

DisbursementID	Integer	Y	FKey reference to Disbursements.DisbursementID column; Disbursement ID applied to the ticket issuance transaction
RSEventSeatMapID	Integer	Y	FK link to RSEventSeatMaps
RSEventSeatID	Integer	Y	FK link to RSEventSeats
SectionName	Char(10)	Y	Name of the reserved seating section
RowName	Char(10)	Y	Name of the row in the reserved seating section
SeatName	Char(10)	Y	Name of the seat within the reserved seating row
JnlTripSummaryID	Int	Y	Link back to Jnl Trip Summary table
RedeemedValue	Float	Y	Points redeemed to purchase this ticket
FirstName	Varchar(30)	Y	First name of the guest associated with this ticket
LastName	Varchar(30)	Y	Last name of the guest associated with this ticket
LoyaltyPoints	Float	Y	The LoyaltyPoints column defines how many points were accrued for this ticket in a transaction.
UpsellStatus	Integer	Y	Indicates if there was no upsell journal record associated with this record, or if this was an upsell selection. Also indicates if upsell choices were presented, but rejected. <sup>5</sup>
UpsellPLU	Varchar(20)	Y	The PLU from the upsell journal record associated with this ticket record. This is the original PLU that that was chosen that caused the user to select this ticket for upsell. This will be blank if this ticket was not added because of an upsell.
UpsellPriceDifference	Money	Y	The price difference from the upsell journal record associated with this ticket record. For replacement upsell, this is the difference in the price between the original PLU chosen, and the price of the upsell PLU. For add-on upsell, this is the full price of the upsell PLU. This is 0.00 if the item is not an upsell item.
UpsellType	Integer	Y	This is the upsell type from the upsell journal record associated with this ticket record. This is the type of upsell that determines how the upsell item was added to the transaction (as a replacement for the chosen item, or added with the original item). <sup>6</sup>
UpsellUserID	Integer	Y	The user ID of the user who performed the upsell. For POS transactions this will be the same as the transaction user. For order transactions, the upsell is performed by the user who adds the upsell items to the order, which is not necessarily the same as the user who issues the transaction.
UpsellSalesChannelType	Integer	Y	The sales channel type indicates where the upsell was performed (POS, OE, kiosk, Web). <sup>7</sup>
GiftAidAmount	Money	Y	The additional amount received for the ticket as a Gift Aid donation. This does not populate for full Gift Aid donations.
GiftAidType	Integer	Y	The type of gift aid sold with the ticket. 0 indicates that there is no gift aid. <sup>8</sup>
ContactID	Int	Y	Foreign key to CustContacts.CustContactID
MemberAddOnID	Int	Y	The unique ID of the joint member add-on referenced by the JnlTicket record. Foreign key to JointMembers.JointMemberID.
EntitlementAddOnVisualID	nvarchar(40)	Y	Visual ID of the package that the ticket is being added to as an entitlement add-on.
PkgInstanceDetailID	Int	Y	Foreign key to PkgInstanceDetails.PkgInstanceDetailID. For returns of tickets, passes, or debit in a package, this contains the ID of the detail from the PkgInstanceDetails table.
PackagePrintSequence	Int	Y	The sequence number indicating the order in which the package detail was printed.
RFIDSerial	nvarchar(20)	Y	Serial number from the RFID chip that the ticket is encoded on.
UpgradeValue	Float	Y	Upgrade value retrieved from the pricing processor plugin.
PriceToken	Varchar(60)	Y	An identifier used to reference the associated external price data for future pricing requests.
Reprinted	Bit	Y	Indicates that the product has already been reprinted.
ActivateByDate	Int	Y	The date by which an inactive ticket must be activated.
EndOfLifeDate	DateTime	Y	The date Galaxy estimates that the ticket will expire. This estimate is done at the time of sale.
EndOfLifeDateStatus	Int	Y	The status of the end of life date. Indicates if the end of life date has been overridden. <sup>8</sup>
EndOfLifeLockWindow	Int	Y	Number of days past the EndOfLifeDate in which the ticket will lock. -1 indicates that the ticket will not lock.
UnitPrice	money	N	The unit price of a ticket. For a regular ticket this will be the same as Price. For a group ticket Price = UnitPrice * Qty.

**Indexes**

Name	Kind	Columns	Purpose
PKJnlTicketsJnlDetailID	P	JnlDetailID	Primary Key.
IXJnlTicketsTransNoNodeNo		TransNo, NodeNo	Index to speedup Ticket Reason Report
IXJnlTicketsEventNo		EventNo	Improve query performance for Resource Management Revenue Performance Report
IXJnlTctsPLUJnlDtIDVID		PLU, JnlDetailID, VisualID	Used by the reseller reports
IXJnlTicketsOrderNo	IX	OrderNo	Improve query performance for selects by OrderNo
IXJnlTicketsDateSold	IX	DateSold	This index speeds up the query to load pass information for SIAE reporting.
IXJnlTicketsVisualID	IX	VisualID	Improve query performance for selects by VisualID
IXJnlTicketsContactID	F	ContactID	Foreign key to CustContacts.CustContactID
IXRevenuePerformanceReport	IX	EventNo, Qty, JnlDetailID, Company, AccessCode, DateSold, DiscNo, Price, Tax, DiscAmt, TktCode	Improve query performance for the Revenue Performance Report
IXTicketUpgradeLookup	IX	TktIndex, PLU, TransNo, NodeNo	Improve query performance for ticket upgrade look-ups

**<sup>1</sup> Ticket Status Values**

Value	Gateway Constant Name	Description
0	TKT_VALID	The ticket is valid.
1	TKT_VOIDED	The ticket has been voided.
2	TKT_RETURNED	The ticket has been returned.
3	TKT_INACTIVE	Inactive ticket
4	TKT_ACTIVATED	Ticket has been activated. This is a value only used in JnlTickets. TCon32 reads this value and sets the ticket status from TKT_INACTIVE to TKT_VALID.
5	TKT_UPGRADED	Ticket was upgraded using the Ticket Upgrade function in Galaxy
6	TKT_REPLACED	Ticket was replaced using the Reprint function in Galaxy
7	TKT_REPRINTED	Ticket was reprinted using the Reprint function in Galaxy, and the "Create New VisualID When Reprinting Tickets" option in Order Entry config is enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain.

**<sup>2</sup> Pass Status Values**

Value	Gateway Constant Name	Description
0	PASS_VALID	Pass is valid.
1	PASS_VOIDED	Pass has been voided.
2	PASS_RETURNED	Pass has been returned.
3	PASS_REPLACED	Pass was replaced (from upgrade, reissue, renewal) or Pass was reprinted using the Reprint function in Galaxy
4	PASS_PURCHASER	This is a purchaser pass.
5	PASS_EXPIRED	Pass is expired.
6	PASS_UPGRADED	Pass was upgraded using the Ticket Upgrade function in Galaxy.
7	PASS_REPRINTED	Pass was reprinted using the Reprint function in Galaxy, and the "Create New VisualId When Reprinting Tickets" option in Order Entry config is enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain.
8	PASS_BLOCKED	Pass is blocked.
9	PASS_UNISSUED	Pass was created as part of an order, but has not yet been issued.
10	PASS_RESTRICTED_ADMISSION	Pass is valid for admission, but is not valid for pass-required tickets or pass-required discounts.
11	PASS_APPROVED	This status corresponds to a pass photo status of Approved when config option is selected to set pass and pass photo statuses to be the same.
12	PASS_NOT REVIEWED	This status corresponds to a pass photo status of Not_Reviewed when config option is selected to set pass and pass photo statuses to be the same.
13	PASS_REJECTED	This status corresponds to a pass photo status of Rejected when config option is selected to set pass and pass photo statuses to be the same.
14	PASS_NOT_PRINTED	The pass exists and IDs have been generated, but it was not printed. This is used for SIAE in Order Entry since pass information must be generated at the time of payment.
15	PASS_INACTIVE	Pass is inactive.

**<sup>3</sup> TaxMethod values, per character**

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis.
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis.
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary.
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis.

**<sup>4</sup> UpdateCode Values**

Value	Gateway Constant Name	Description
0	UPDATECODE_NORMAL	Normal case, ticket was added before it was used.
1	UPDATECODE_UNSOLD	The ticket was not found when first attempting to use it, and was overridden to admit the guest anyway. Doing this resulted in TCON32 adding this ticket dynamically. However, the sales data for this ticket has not yet been received. Therefore, many of the column values for this ticket are blank or zero or NULL (PLU, Price, DateSold, etc.).
2	UPDATECODE_UPDATED	The ticket was created dynamically with an UpdateCode value of 1, but eventually the late sales data was received. All column values were updated with the values from the sales data, except for those used to track the usage of the ticket (UseCount, RemainingValue, etc.).
3	UPDATECODE_OLD_USE	Similar to value 2, the ticket was created dynamically and updated with late sales data. However, the DateSold time is after the first use of the ticket, indicating a possible mismatch between the clocks of TCON32 and the POS, or a counterfeit ticket may have been used.

**<sup>5</sup> UpsellStatus Values**

Value	Gateway Constant Name	Description
0	UPSELL_STATUS_NO_UPSELL	No upsell record exists for this ticket
1	UPSELL_STATUS_SELECTED	This ticket was created as the result of an upsell
2	UPSELL_STATUS_CANCELLED	Upsell options were presented for this ticket, but were rejected.
3	UPSELL_STATUS_TRANSACTIONAL	This item was sold as part of a transactional upsell.

**<sup>6</sup> UpsellType Values**

Value	Gateway Constant Name	Description
0	UPSELL_TYPE_REPLACEMENT	This ticket was added to replace the original PLU selected
1	UPSELL_TYPE_ADD_ON	This ticket was added in addition to the original PLU selected

**<sup>7</sup> UpsellSalesChannelType**

Value	Gateway Constant Name	Description
0	sctPOS	The upsell was performed from the Point of Sale
1	sctOrderEntry	The upsell was performed from Order Entry
2	sctKiosk	The upsell was performed from the Kiosk
3	sctWebStore	The upsell was performed from the WebStore

**<sup>8</sup> GiftAidType Values**

Value	Gateway Constant Name	Description
0	gaNone	No gift aid is associated
1	gaFixedAmount	Gift Aid amount holds the amount of Gift Aid for the ticket
2	gaFull	Gift Aid was used for the full amount of the ticket
3	gaNonGAFixedAmount	Gift Aid amount is for a non-Gift Aid donation for the ticket
4	gaNonGAFull	A non-Gift Aid donation was made for the full amount of the ticket

**<sup>8</sup> EndOfLifeStatus Values**

Value	Gateway Constant Name	Description
0	EOL_STATUSCODE_DEFAULT	Default status for the end of life date
1	EOL_STATUSCODE_OVERRIDDEN	The end of life date has been manually overridden

## 12.56 JnlTranRefunds

This table stores transportation refund journal information.

### Columns

Column	Type	Allow Nulls	Description
JnlTranRefundId	integer	N	
JnlTranID	integer	N	
Company	integer	N	
Category	integer	Y	
SubCategory	integer	Y	
OrigAgency	integer	Y	
OrigDate	datetime	Y	
Ticket	integer	Y	
Coupons	integer	Y	
Name	char(30)	Y	Refund Name
Address	char(30)	Y	Street Address
Address2	char(30)	Y	City and State Address
Phone	char(20)	Y	Phone Number
Reason	char(30)	Y	Reason for Refund

### Indexes

Name	Kind	Columns	Purpose
PKJnlTranRefundID	PK	JnlTranRefundID	Primary Key
CXJnlTranRefunds	CX	JnlTranID, JnlTranRefundID	

## 12.57 JnlTripSummaries

Designed to work with the Transportation module, this table stores the summary information for a given trip sold.

### Columns

Column	Type	Allow Nulls	Description
JnlTripSummaryID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to Journal Detail
RouteType	Char(2)	N	This is either a One Way (OW) or Return (RT)
FareClass	Char(2)	N	This is defined on the Transportation Templates and example are AD, SC, CH which corresponds to the ticket types. i.e. Adult, Senior, Child.
Origin	Int	N	This is the Tickets origin city
Destin	Int	N	This is the Tickets final destination city.
TariffMilesOB	Int	N	This is the total tariff miles outbound that is calculated by adding up the tariff Leg miles from the routing record(s)
TariffMilesIB	Int	Y	This is the total tariff miles inbound that is calculated by adding up the tariff Leg miles from the routing record(s)
SchedMilesOB	Int	N	This is the total schedule miles outbound that is calculated by adding up the schedule Leg miles from the routing record(s)
SchedMilesIB	Int	Y	This is the total schedule miles inbound that is calculated by adding up the schedule Leg miles from the routing record(s)
DepartDate	DateTime	N	This is from the selected itinerary
ReturnDate	DateTime	Y	This is from the selected return itinerary
DepartTime	DateTime	Y	From selected Itinerary (schedule times should not be looked up later since schedules can change). This field does not have a date, just the time.
ReturnTime	DateTime	Y	From selected Itinerary (schedule times should not be looked up later since schedules can change). This field does not have a date, just the time.
TicketType	Char(16)	Y	From the selected fare ticket type description
ItemPLU	Char(20)	Y	Galaxy PLU of the transportation ticket
ICWPLU	Char(20)	Y	Galaxy PLU of the In-Conjunction With item
TariffCarrier	Char(4)	Y	Refers to Tariffs.Carrier
TariffName	Char(24)	Y	Refers to Tariffs.Name. A tariff is uniquely identified by Carrier + TariffName. We cannot use TariffID (record ID) since ID values may change during a backup/restore or by simply deleting and re-entering the record.
InboundDepartTime	DateTime	Y	From selected Itinerary. This field does not have a date, just the time.
ArriveTime	DateTime	Y	From selected Itinerary. This field does not have a date, just the time.

### Indexes

Name	Kind	Columns	Purpose
PKJnlTripSummaries	P	JnlTripSummaryID	Primary Key.
CXJnlTripSummaries	I	JnlTranID, JnlTripSummaryID	Clustered Index for transaction lookups

## 12.58 JnlUDFFields

This table contains User Defined Field information for a transaction.

### Columns

Column	Type	Allow Nulls	Description
JnlUDFFieldID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID.
UserFieldID	Integer	N	OEUserFields.OEUserFieldID
Response	Varchar(40)	Y	User response
UDFType	Int	Y	Type of UDF

### Indexes

Name	Kind	Columns	Purpose
PKUDFFieldID	P	JnlUDFFieldID	Primary Key (non-clustered).
CXUDFFieldsUDFFieldID	CX	JnlTranID, UDFFieldID	Clustered Index.  Used to load a list of records from the JnlUDFFields table where the JnlTranID falls within a specified range.

**12.59 JnlUpgrades**

This table holds the previous information for an upgraded ticket.

**Columns**

Column	Type	Allow Nulls	Description
JnlUpgradeID	Int	N	Primary key, always unique
JnlTranID	Int	N	Journal transaction ID
VisualID	Varchar(40)	N	Previous VisualID
Price	Money	N	Previous Ticket's Price
Tax	Money	N	Previous Ticket's Tax
NewVisualID	Varchar(40)	Y	Visual ID of the new upgraded ticket
NetPrice	Money	Y	Difference between original ticket upgrade return value and upgrade ticket price
NetTax	Money	Y	Difference between original ticket upgrade return tax and upgrade ticket tax
RetainMedia	Bit	Y	Original media was retained during upgrade

**Indexes**

Name	Kind	Columns	Purpose
PKJnlUpgradesJnlUpgradeID	P	JnlUpgradeID	Primary Key.
CXJnlUpgradsJnlTrnIDJnlUpgrdID		JnlTranID, JnlUpgradeID	Clustered Index. Used to load a list of records from the JnlUpgrades table where the JnlTranID falls within a specified range.

## 12.60 JnlUpsells

The JnlUpsells table provides supplementary information for the corresponding JnlDetails record when that detail refers to an upsell record. The JnlCodeID value of the corresponding record in the JnlDetails table is 1016.

### Columns

Column	Type	Allow Nulls	Description
JnlUpsellID	Integer	N	Primary key, always unique.
JnlTranID	Integer	N	Foreign key to JnlHeaders.JnlTranID
PLU	VarChar(20)	N	PLU of the item that was upsold from. This will be blank if the upsell was cancelled.
Quantity	Integer	N	Quantity of the item required for upsell. For item-based upsell, this value is always 1. For transactional upsell, this is the quantity of the item that is replaced when the upsell occurred.
PriceDifference	Money	N	The difference in price between the original PLU selected, and the upsell item chosen. This is the full price of the item for Add-On upsell options. For transactional upsell, this value is always 0, since the price difference is stored in a journal record summarizing the price difference for all requirements.
UpsellType	Integer	N	Type of upsell. This indicates if the upsell was added to the original item, or replaced the original item selected. <sup>1</sup>
UserID	Integer	N	ID of the user that performed the upsell.
SalesChannelType	Integer	N	The sales channel type indicates where the upsell was performed (POS, OE, kiosk, Web). <sup>2</sup>

### Indexes

Name	Kind	Columns	Purpose
PKJnlUpsellID	P	JnlUpsellID	Primary key - Unique ID in the table.
CXJnlUpsellsJTDJnlUpsellID	A	JnlTranID, JnlUpsellID	Clustered index.  Used when retrieving upsell data for a transaction.

### <sup>1</sup> UpsellType Values

Value	Gateway Constant Name	Description
0	UPSELL_TYPE_REPLACEMENT	The PLU was added as a replacement for the original PLU chosen.
1	UPSELL_TYPE_ADD_ON	The PLU was added to the transaction in addition to the original option chose.

### <sup>2</sup> SalesChannelType Values

Value	Gateway Constant Name	Description
0	sctPOS	The upsell was performed from the Point of Sale
1	sctOrderEntry	The upsell was performed from Order Entry
2	sctKiosk	The upsell was performed from the Kiosk
3	sctWebStore	The upsell was performed from the WebStore

## 12.61 JnlUsage

This table contains usage information from local journal.

Note: This table does not contain the RFC usage information.

### Columns

Column	Type	Allow Nulls	Description
JnlUsageID	int	N	Primary key, always unique
JnlTranID	int	N	Foreign key to JnlDetails and JnlHeaders JnlTranID
Code	int	N	The type of activity Valid values for this column are shown in the <i>Code Values</i> table. Refer to Usage.Code.
UseTime	datetime	N	Usage time of the ticket or scan. This contains Date and Time with milliseconds.
ACP	int	N	The ID number of the Access Control Point at which the activity took place.
AccessCode	int	N	The AccessCode extracted from the barcode scan which generated the Usage record.
Status	int	N	The result of the activity, as defined in the <i>Status Values</i> table. Refer to Usage.Status.
Quantity	int	N	The number of guests admitted or value redeemed.
UseNo	int	N	The number of time the ticket or pass was used.
Operator	int	N	Foreign key into GXUers.UserID, referencing the user logged on at the time the Usage record was created. This value is 0 if nobody was logged-on.
EntryMethod	int	N	A number representing the method used to enter the barcode into the system which in-turn generated the Usage record. Possible values of EntryMethod are defined in the <i>EntryMethod Values</i> table below  Refer to Usage.EntryMethod
Override	int	N	If the operator overrides an invalid ticket and lets the guest into the park, this field stores the original reason the ticket was rejected <sup>1</sup> .  Refer to Usage.Override
UsageCondition	int	N	A number representing the condition of TCon32 when the ticket was scanned. Possible values of UsageCondition are defined in the <i>UsageCondition Values</i> table below.  Refer to Usage.UsageCondition
BankNo	int	N	The bank detail number validated by ACS2. In other words, the bank detail that was valid and allowed the admission. This bank number is sent to TCON32 which uses it to determine the correct ticket bank detail to remove uses from.
Scan	Varchar(40)	N	
OperationID	Int	N	Operation that was used
BiometricStatus	Int	N	Status of Biometric activity performed during this usage  Refer to Usage.BiometricStatus
SerialStart	Int	N	Serial number starts at this position in Scan
SerialLen	int	N	Length of serial number
AdultQty	Int	Y	The number of adults admitted on the validation. (From the Usage2 journal record)
ChildQty	Int	Y	The number of children admitted on the validation. (From the Usage2 journal record)
ScannedVisualID	Varchar(40)	Y	The visual ID that was scanned for the membership. If a superticket visual ID is scanned, this will contain that visual ID. The visual ID on the usage will be replaced by the visual ID of the referenced ticket or pass.
GuestQty	Int	Y	Stores the number of guests admitted through an Admission Control Point with a Joint Membership Pass.
JointMemberID	Int	Y	FK to JointMembers.JointMemberID. This indicates the JointMember that the usage was generated for.
AttractionID	Int	Y	The Attraction that was configured on the ACP where this usage occurred. Foreign key to Attractions.IDNo.
OriginalUsageID	Int	Y	The original usage id that this usage was based off of. Used for Reversals and Voids of usages.
FacilityID	Int	Y	Foreign key to Facility.IDNo. ACPS can now function as an exit of a facility. If zero, the facility that the usage is applied to can be obtained via Usage.ACPS. If non-zero, the usage applies to the indicated facility.
EntitlementCharged	Bit	Y	If 1, the entitlement has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION.
BankCharged	Bit	Y	If 1, the bank has been charged for the admission on a different usage. This may be set to 1 on an attraction usage if a guest is admitted to a facility with status FACILITY_ADMISSION and admitted to an attraction with status ATTRACTION_ADMISSION, and the same bank was used for both admissions.
NodeNo	Int	Y	The node number this activity occurred at.
TransNo	Int	Y	The POS transaction number this activity occurred in (if available).
PLU	nVarChar	Y	The PLU of the scanned item (if available).
ExternalUsage	bit	Y	Indicates if the usage originated from an external validation.
ACSRReservationConfirmationNumber	nvarchar(50)	Y	The confirmation number of the reservation used during validation.
UsageGUID	UniquedIdentifier	Y	GUID associated to this JnlUsage. Use for joining a JnlUsage row to a Usage row.

### Indexes

Name	Kind	Columns	Purpose
PKJnlUsageID	P	JnlUsageID	Primary Key.
CXJnlUsageJnlTranIDJnlUsageID	CX	JnlTranID, JnlUsageID	Unique Clustered Index
IXJnlUsageUsageGUID	IX	UsageGUID	Improve performance when finding matching JnlUsage and Usage rows.

## 12.62 JnlVoid

The **JnlVoid** table provides supplementary information for the corresponding **JnlDetails** record when that detail refers to a void record. The **JnlCodeID** value of the corresponding record in the **JnlDetails** table is 4.

### Columns

Column	Type	Allow Nulls	Description
JnlVoidID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to <b>JnlHeaders.JnlTranID</b> .
Commission	Float	N	Commission amount.
VoidTranNo	Integer	N	Transaction number that was voided.
VoidJnlTranID	Integer	N	Foreign key to <b>JnlHeaders.JnlTranID</b> specifying the matching transaction for the same node.
SupervisorID	Integer	Y	Supervisor ID used for supervisor approval for a voided transaction.

### Indexes

Name	Kind	Columns	Purpose
PKJnlVoid	P	JnlVoidID	Primary Key.
CXJnlVoid		JnlTranID, JnlVoidID	Clustered Index. Used to load a list of records from the <b>JnlVoid</b> table where the <b>JnlTranID</b> falls within a specified range.

## 12.63 JnlVouchers

The **JnlVouchers** table contains information about Voucher transactions recorded in the journal. This table will contain records only if using the Vouchers module. The JnlCodeID value of the corresponding record in the **JnlDetails** table is 51.

### Columns

(None)

Column	Type	Allow Nulls	Comments
JnlVoucherID	Integer	N	Primary key, always unique.
JnlTranID	Integer	Y	Foreign key to JnlHeaders.JnlTranID.
VoucherID	Integer	Y	Foreign key to Vouchers.VoucherID, specifying the Voucher processed in the transaction.
IssuerCode	Char(3)	Y	Corresponds to IssuerCode field of Vouchers table. <sup>1</sup>
ArrivalDate	Datetime	Y	Corresponds to ArrivalDate field of Vouchers table. <sup>1</sup>
Serial	Char(6)	Y	Corresponds to SerialNumber field of Vouchers table. <sup>1</sup>
Status	Integer	Y	Status of voucher after the transaction. <sup>2</sup>
Decreased	Bit	N	Indicates whether or not a decrease certificate was printed.

### Indexes

Name	Kind	Columns	Purpose
PKJnlVouchersJnlVoucherID	P	JnlVoucherID	Primary Key.
CXJnlVouchersJnlTranIDJnlVchrID		JnlTranID, JnlVoucherID	Clustered Index.  Used to load a list of records from the JnlVouchers table where the JnlTranID falls within a specified range.

<sup>1</sup> The IssuerCode, ArrivalDate, and Serial fields can be used together to uniquely identify Voucher records.

<sup>2</sup> Status Values

Value	Gateway Constant Name	Description
0	vsImported	Imported (Ok to be redeemed)
1	vsIssued	Issued (Tickets on the voucher have been redeemed)
2	vsCanceled	Cancelled (All tickets on this voucher has been cancelled)
3	vsDynamic	Dynamically created
4	vsPrePrinted	Pre-Printed
5	vsPickedup	Reserved (not currently used)
6	vsEditing	Dynamically created voucher currently being edited

**12.64 JournalBackupLog****Columns**

Column	Type	Allow Nulls	Description
JnlLogID	Int	N	
NodeNo	Int	N	
FileType	Char(1)	N	
FileName	Char(40)	N	
Complete	Bit	N	

**Indexes**

(none)

## 12.65 MemoTbl

The **MemoTbl** table stores memo information created by Galaxy.

### Columns

Column	Type	Allow Nulls	Description
MemoTblID	Int	N	Primary key, always unique.
MemoType	Char(8)	N	Describes type of memo. <sup>1</sup>
OwnerId	Int	N	ID of record having notes or 0 for journal memos. <sup>1</sup>
MemoLineNo	Int	N	Always has value 1 (reserved for possible future use).
MemoText	Text	N	Contains the memo text.

### <sup>1</sup> MemoType Values

Value	Gateway Constant Name	Description
0-999		Journal memos for Transaction Prompt responses - Value is the "ID" specified in the "Transaction Prompt" section of the General Configuration
9		Journal memos for misc. memos without an ID
1000	NEW_CUST_MEMO	Journal memos for new customers
1001	NEW_PASS_MEMO	Journal memos for new passes
1002	REISSUE_PASS_MEMO	Journal memos for reissued passes
1003	CARD_NAME_MEMO	Journal memos for card names of IATA Network travel agent ID cards
1004	OVERPAYMENT_MEMO	Journal memos for overpayments
1005	ADJUSTMENT_MEMO	Journal memos for adjustments
1006	REPRINT_MEMO	Journal memos for order ticket reprints
1007	SURVEY_MEMO	Journal memos for surveys
1008	STARTUP_MEMO	Journal memos for recording versions at Startup
1009	EXPORT_MEMO	
1010	IMPORT_MEMO	
1011	PASS_NOTES_MEMO	Journal memos for pass notes
1012	TICKET_DATE_MEMO	
1013	PASS_DISC_MEMO	
1014	CORRECTION_MEMO	
1015	CARD_TYPE_MEMO	
1016	AUTH_CENTER_MEMO	
1017	AUTH_BANK_DESC_MEMO	
1018	CANCELLED_PAYMENT_MEMO	
1019	ACCOUNT_VERIFICATION_MEMO	
1020	TRANSACTION_NOTE_MEMO	A single transaction note can be entered for each transaction in the system.
1021	VEHICLE_ENTRY_MEMO	Vehicle entry memo record, created when a vehicle detection loop is installed
1022	VEHICLE_EXIT_MEMO	Vehicle exit memo record, created when a vehicle detection loop is installed
1023	LINK_PASS_MEMO	
1024	FAILED_ACTIVATION_MEMO	
1025	ICW_MEMO	
1026	MANUAL_MEMO	
1027	CCC_EFT_SETTLEMENT_TYPE_MEMO	
1028	CCC_MERCHANT_ID_MEMO	
1029	CCC_TERMINAL_ID_MEMO	
1030	CCC_MESSAGE_MEMO	
1031	CCC_TRANS_REF_NUM_MEMO	
1032	MAGIC_BATCH_NO_MEMO	MagIC protocol settlement batch number
1033	MAGIC_BATCH_COUNT_MEMO	MagIC protocol settlement batch count
1034	MAGIC_BATCH_AMOUNT_MEMO	MagIC protocol settlement batch amount
1035	MAGIC_BATCH_FAILED_MEMO	MagIC protocol settlement batch failed
1036	MAGIC_HOST_NO_MEMO	MagIC protocol host number for settlement
1037	TOGGLE_SURVEYS_OFF_MEMO	Toggle surveys off memo record. Indicates that survey prompting was disabled for the POS transaction.
1038	TOGGLE_SURVEYS_ON_MEMO	Toggle surveys on memo record. Indicates that survey prompting was enabled for the POS transaction.
1039	TOGGLE_VEHICLE_DETECTOR_BYPASS_ON_MEMO	Toggle vehicle detector loop bypass on memo record. Indicates that the vehicle detector loop bypass was turned on for the transaction.
1040	TOGGLE_VEHICLE_DETECTOR_BYPASS_OFF_MEMO	Toggle vehicle detector loop bypass off memo record. Indicates that the vehicle detector loop bypass was turned off for the transaction.
1041	ENABLE_VEHICLE_DETECTOR_MEMO	
1042	DISABLE_VEHICLE_DETECTOR_MEMO	
1043	REINITIALIZE_VEHICLE_DETECTOR_MEMO	
1044	TRANSACTION_SCAN_ATTRIBUTE_MEMO	Journal memos for External Usage Attributes. Value of OwnerId is a foreign key reference to JnlUsage.JnlUsageId.
1045	MODIFIER_SPECIAL_INSTRUCTION_MEMO	Holds custom text from a special instruction food and beverage modifier.
1046	PRICE_TOKEN_MEMO	Identifier used for referencing external pricing data.
1047	BOOKING_REFERENCE_MEMO	Booking reference data recorded for a Reseller transaction.
1048	TRAVEL_DATE_MEMO	Travel date recorded for a Reseller transaction.
1049	FISCAL_ARCHIVE_FILE_HASH_MEMO	Hash of the output file for fiscal archives.
1050	FISCAL_ID_MEMO	The Company Fiscal ID associated with a transaction.

1051	IJOURNAL_LOG_MEMO	Journal memos for log messages.
1052	CHANGE_MODE_MEMO	
1053	ORCA_OFFLINE_TRANSACTION_MEMO	An audit trail of transaction and usage changes when processing ORCA offline transactions.

**Indexes**

Name	Kind	Columns	Purpose
PKMemoTblMemoTblID	P	MemoTblID	Primary Key.
IXMemoTblMemoType		MemoType	Used by DbSynch to gather lists of memo records specific to a certain type (STARTUP_MEMO).

## 12.66 SequenceNumbers

It's used to store ticket / pass sequence numbers. The numbers are stored by node and company since each node can produce tickets for multiple companies. Currently, the table is only used by eGalaxy (when assigning a VisualID to a ticket) but will likely be used for POS machines in the future.

### Columns

Column	Type	Allow Nulls	Description
SequenceNumberID	Int	N	Primary key, always unique
Node	Int	Y	The node number for which this chain of sequence numbers is used for.
Company	Int	Y	Each node can use sequence numbers for multiple companies, each company sequence number starts at 1
SequenceNumber	Int	Y	The actual sequence number. The value is the LAST number used by a node for the specified company.

### Indexes

Name	Kind	Columns	Purpose
PKSequenceNumberID	P	SequenceNumberID	Primary Key.
IXSequenceNumberNodeCompany	IX	Node, Company	Used to query for a unique sequence number.

## 12.67 Settlements

The **Settlements** table is used to store credit card settlement information, supplementing the **JnlDrafts** table. No settlement information is journalized for some credit card protocols. If using a protocol that does journalize settlement records, there is normally one settlement record for each authorization (except for VISANET2, which has two settlement records for each authorization).

### Columns

Column	Type	Allow Nulls	Description
SettlementID	Int	N	Primary key, always unique.
JnlDraftID	Int	N	Foreign key to JnlDrafts.JnlDraftID, referencing the authorization this settlement information applies to.
JnlCodeID	Int	N	41 for most credit card protocols, or 46.
SettlementCode	Char(31)	N	Main information needed for settlement. <sup>1</sup>
SortCode	Char(16)	N	Contains extra settlement information if JnlCodeID is 46. <sup>1</sup>
Sequence	Char(4)	N	Contains extra settlement information if JnlCodeID is 46. <sup>1</sup>
OrigSaleDate	DateTime	Y	Contains the original sale date of returns and voids, if JnlCodeID is 41.

### Indexes

Name	Kind	Columns	Purpose
PKSettlementsSettlementID	P	SettlementID	Primary Key.

<sup>1</sup> The data in these columns are returned by the host (not created by the system), the format depends upon the credit card protocol being used.

## 12.68 ShiftStatus

Contains one record for each shift started with a Change Fund journal record.

The shift is inserted in this table when the Change Func record is DBSynched.

The status is updated to Suspend when the SUSPEND journal record is DBSynched.

The status is updated to Cashed Out by the Remote Cashout process or when the CASHOUT journal record is DBSynched.

For shifts cashed out locally (not using the Remote Cashout process) the Status will be updated only by DBSynch.

The RemoteCashoutID is updated by the Remote Cashout process or when the CASHOUT\_SHIFT\_REC is DBSynched.

Column	Type	Allow Nulls	Description
ShiftStatusID	Int	N	Primary key, always unique
ShiftNo	Int	N	Shift Number
NodeNo	Int	N	Node Number
UserID	Int	N	User identification
AgencyNo	Int	N	Agency Number
ShiftStart	Datetime	N	Date/Time of the beginning of the shift.
ShiftEnd	DateTime	Y	Date/Time of the end of the shift.
FromTranNo	Int	N	The transaction number of the change fund record that starts a new shift.
ThruTranNo	Int	Y	The transaction number of the suspend or the cashout record
Status	Int	N	At the insert time the record will have the status set as 1 (OPEN). When a SUSPEND transaction is DBSynched, the status will change to 2 (SUSPENDED). When a CASHOUT transaction is found, the status of the node from which the cashout is processed, will change to 3 (CASHED-OUT). This means that the suspended shifts will never have their status changed to cashed out. The RemoteCashoutID field determines if that shift had already been included in a cashout or not. <sup>1</sup>
RemoteCashoutID	Int	Y	Identify the shifts that were remotely cashed out together. All the shifts cashed out together have the same RemoteCashoutID <sup>2</sup> . -1 indicates that the shift has not been cashed out yet. It didn't take part on a remote cashout and was not cashed out locally either. 0 indicates that the shift was cashed out locally (in opposite to remote cashout). As the local cashout includes de data of ONE SHIFT ONLY the status of the local cashed out shifts will be 3. Any value greater than 0 indicates that the shift was remotely cashed out and all the shifts with this same remote cashout id number were cashed out together (in the same remote cashout process).

## Indexes

Name	Kind	Columns	Purpose
PKShiftStatusShiftStatusID	P	ShiftStatusID	Primary Key.
AXShiftStatusNodeNoFromTranNo	A	NodeNo, FromTranNo, ShiftStatus	Unique index used to update the ShiftStatus table after exporting all the journal records. NodeNo, FromTranNo, and ShiftStart make a row unique
IXShftStNodUsrAgncStatStrtThru		ShiftNo, NodeNo, UserID, AgencyNo, Status, ShiftStart, ThruTranNo	Index used to update the ShiftStatus table during the journal export for suspend shift and cashout shift journal records.
IXShiftStatusShiftStart		ShiftStart	Index used to gather all the shifts for a user on a remote cashout.
IXShiftStatusNodeUserFromTranNo		NodeNo, UserID, Status, ShiftStatusID, FromTranNo	Used when loading the current open shift.

## <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	OPENED_SHIFT	Open (initial value)
2	SUSPENDED_SHIFT	Suspended (After the SUSPEND journal transaction is exported)
3	CASHED_OUT_SHIFT	Cashed out (After cashout process of after the CASHOUT journal transaction is exported)

## <sup>2</sup> RemoteCashoutID Values

Value	Gateway Constant Name	Description
-1	NOT_CASHED_OUT	The shift has not been cashed out yet.
0	CASHED_OUT_LOCALLY	This shift was cashed out locally (in opposite to remote cashout)
> 0	N/A	This shift took part in a remote cashout. All the shifts with this same remote cashout id were cashed out together.

**13 Kiosk / Web Store / Sales Channel**

### 13.1 CategoryPriceData

The CategoryPriceData table is used by the eGalaxy web store to quickly access pricing by category for web page display purposes. This table is generated and sent during a publish to the eGalaxy web store.

#### Columns

Column	Type	Allow Nulls	Description
CategoryPriceDataID	Int	N	Primary key, always unique
CategoryID	Int	N	FK to SalesChannelDetails.SalesChannelDetailID when SalesChannelDetails.DetailType = 1(ID_SALES_CAT_DETAIL), the Category that this entry relates to.
PLU	Char(20)	N	FK to Items.PLU, the ticket/item that this entry relates to.
Price	Money	Y	The price for 1 ticket/item as calculated by Galaxy.
Discount	Money	Y	The discount amount for 1 ticket/item as calculated by Galaxy.
Tax	Money	Y	The tax amount for 1 ticket/item as calculated by Galaxy.

#### Indexes

Name	Kind	Columns	Purpose
PKCategoryPriceDataID	P	CategoryPriceDataID	Primary Key
IXCategoryPriceDataCategoryIDPLU		CategoryID, PLU	Speed up lookups by CategoryID and PLU

### 13.2 MarketingMessages

#### Columns

Column	Type	Allow Nulls	Description
MarketingMessageID	Int	N	Primary key, always unique.
Template	Text	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKMarketingMessageID	P	MarketingMessageID	Primary Key.

### 13.3 Merchants

A Merchants table contains details specific to a merchant. Galaxy CustomerID, SurveyID to use for this merchant etc. are defined in this table. The table is used by the web store to get details for a given merchant. The name and address of the merchant can be found by looking up the CustomerID in the Customers table.

Column	Type	Allow Nulls	Description
MerchantID	Integer	N	Primary key. Obtained from GatewayCounters.
Code	Varchar (30)	N	Merchant identifier
Description	Text	Y	Description about the merchant
CustomerID	Integer	N	Galaxy Customer ID to use for this merchant
SurveyID	Integer	Y	The SurveyID to use when taking surveys at the time of completing an order.
Logo	VarChar(256)	Y	URL to Merchant logo image
SuppressMarketingMessages	Bit	Y	If set to 1, marketing messages defined on any detail of sales channel or shipping method are not displayed on both confirmation web page and confirmation e-mail  If set to 0, which is default no restrictions are applied. Marketing messages defined on the any detail of sales channel and shipping method  are displayed if shipping method is configured to do so
WebPageHeaderTemplate	Varchar (256)	Y	Name of the header web page template to use for all web pages displayed after this merchant is selected
PromptForEmailNewsletter	Bit	Y	If set to 1, the web store prompts guest with question "If the guest wants to receive e-mail newsletter or not?"  If set to 0, the web store does not prompt the guest
WebPageFooterTemplate	Varchar (256)	Y	Name of the footer web page template to use for all web pages displayed after this merchant is selected
CustomerSupportURL	VarChar(256)	Y	URL for customer support
PreventBookMarking	Bit	Y	If enabled, you cannot bookmark in a browser
RedirectURL	VarChar(256)	Y	URL to redirect to
WebUseLandingPage	Bit	Y	Web use landing page flag
WebAutoGenerateNavigation	Bit	Y	Web auto generate navigation flag
WebDisplayTermsAndConditions	Bit	Y	Web display terms and conditions
WebTheme	VarChar(128)	Y	Web theme
WebDisplayTermsAndConditionsInLine	Bit	Y	Web display terms and condition in line flag
WSLocalizationGroupID	Int	Y	Foreign key to WSLocalizationGroups table.  This allows Merchant's site to define its own text (wording) on the various system defined text
UsageSurveyID	Int	Y	The SurveyID to use when the customer opts to take the survey at some point after the first use of a ticket from the order.
DisplayEmailOptIn	Bit	Y	Determine if Email Opt-In will be displayed.
DisplayMailingOptIn	Bit	Y	Determine if Mailing Opt-In will be displayed.
DisplaySMSOptIn	Bit	Y	Determine if SMS Opt-In will be displayed.
PreventBookmarkingKind	Int	Y	Method used to prevent unwanted bookmarking of links to the consumer webstore <sup>1</sup>
PreventBookmarkingReferrer	NVarChar(30)	Y	Referrer value (if required by PreventBookmarkingKind) for prevent bookmarking method
PreventBookmarkingDuration	Int	Y	The duration of time in minutes that the time encoded URL is valid for
PreventBookmarkingKey	NVarChar(30)	Y	The value is used to encode the URL when preventing book marking
MerchantGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKMerchantsMerchantID	P	MerchantID	Primary Key
IXMerchantsCode		Code	Used by the query to get merchants based on the code

<sup>1</sup> PreventBookmarkingKind Values

Value	Gateway Constant Name	Description
0	PREVENTBOOKMARKINGKIND_STANDARD	Use the Galaxy standard method
1	PREVENTBOOKMARKINGKIND_TQS1	Use the TQS1 method, requires the PreventBookmarkingReferrer value to be defined

### 13.4 PassBenefits

The purpose of this table is to hold the associated PLU's for each Membership Type. This PLU list is what will be presented to the user as Benefit Tickets at the Kiosk. The PLU marked as IsAutoItem will be added to the transaction as the Member Ticket.

#### Columns

Column	Type	Allow Nulls	Description
PassBenefitID	Int	N	Primary key, always unique
PassKindID	Int	N	Link to SQL PassKinds table
PLU	Char20	Y	Item PLU
Quantity	Int	Y	Maximum number of benefit tickets that can be presented to the user per frequency value
Frequency	Int	Y	0 used for POS, 1 and 2 used for Kiosk. Used with Quantity field. <sup>1</sup>
Description	Char20	Y	Description shown on Kiosk
IsAutoItem	Bit	Y	The PLU the Member will receive for each transaction
Sequence	Int	Y	Display order

#### Indexes

Name	Kind	Columns	Purpose
PKPassBenefitID	P	PassBenefitID	Primary Key.
IXPassbenefitsPassKindID	F	PassKindID	Load benefits by PassKindID

<sup>1</sup> Frequency Values

Value	Gateway Constant Name	Description
0		None
1		Per Day, specifies the Max Qty is on a per day basis
2		Per Transaction, specifies the Max Qty is for every transaction

### 13.5 Promotions

Details of a promotion are defined in the Promotions table. The promotion validity dates, Sales Program to use for applying the rates, Logo and Sample Image etc. are defined in this table. This table is used by the web store to get details for promotions for a given merchant.

Column	Type	Allow Nulls	Description
PromotionID	Int	N	Primary key. Obtained from GatewayCounters.
Name	VarChar(80)	N	Name of the promotion
ExternalID	VarChar(30)	Y	ID used by Web Store to reference this table instead of the UniqueID since the UniqueID could change
Description	VarChar(256)	Y	Description about the promotion
LongDescription	Text	Y	Long description about the promotion
Logo	Image	Y	Logo for the promotion
AlternateLogolmageText	VarChar(512)	Y	Text to be displayed when the Logo image cannot be displayed.
SampleImageHelpText	VarChar(80)	Y	Sample image help text
SampleImage	Image	Y	Sample image to display on internet to helping the guest to find the promotion code within a coupon
AlternateSampleImageText	VarChar(512)	Y	Text to be displayed when the Sample Image cannot be displayed.
IntroText	Text	Y	Intro text to display when this promotion is displayed
CodeEntryPageTitle	Text	Y	Title to display on the promotion code entry page
CodeEntryFieldCaption	Text	Y	Title to display if Code required for entry flag is enabled
ValidationFieldCaption	Text	Y	Title to display if code validation is required to use this promotion
CodeRequiredForEntry	Bit	N	If set to 1, a promotion code is required to see tickets for this merchant  If set to 0, promotion code is NOT required to see tickets
ValidationRequired	Bit	N	If set to 1, promotion has validations and the guest must meet all validations  If set to 0, validations don't exists
ValidFrom	DateTime	N	Promotion start time
ValidThru	DateTime	N	Promotion end time
PromotionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### NOTES:

- Data entered in the fields Description may not be displayed properly on the web store. The size of the field is much larger than what can be displayed on the web store. This is because the user can enter the HTML code within these fields to indicate the font color or font style etc. within the same field
- Fields Long Description, IntroText, CodeEntryFieldCaption, CodeEntryPageTitle, CodeEntryFieldCaption, and ZIPCodeEntryFieldCaption has a data type of Text and therefore these fields can have HTML code and images in it.

#### Indexes

Name	Kind	Columns	Purpose
PKPromotionID	P	PromotionID	Primary key.
XPromoPromoIDValidFromThru		PromotionID, ValidFrom, ValidThru	Used for query to get all promotions for a given Merchant (via PromotionID) and for a given Validity date range

### 13.6 PromotionCodes

A PromotionCodes table contains all codes that the guest can enter to actually use a promotion. The number of times a promotion code can be used is configured here.

Column	Type	Allow Nulls	Description
PromotionCodeID	Int	N	Primary key. Obtained from GatewayCounters.
PromotionID	Int	N	FK reference to Promotions table
Code	VarChar(50)	N	Promotion code
MaxUses	Int	N	Maximum number of times this promotion code can be used  If -1, unlimited number of time  If 0, no uses left for this promotion code  If 1..n, Can be used 1...n times, decremented each time it comes through order
PromotionCodeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKPromotionCodeID	P	PromotionCodeID	Primary key.
IXPromotionCodesCode		Code	Used for query to find get all details based on the code

### 13.7 PromotionOffers

The Promotion Offer consists of a Merchant and a Promotion. The Name, ShortDescription, Description, MerchantID, PromotionID, Logo, SampleImageHelpText, SampleImage, and FopSetID can be defined in this table.

Column	Type	Allow Nulls	Description
PromotionOfferID	Int	N	Primary key. Obtained from GatewayCounters.
Name	VarChar(80)	N	Name of the promotion offer. Data entered here is not displayed on the Web store
ExternalID	VarChar(30)	Y	ID used by Web Store to reference this table instead of the Unique ID since the UniqueID could change
ShortDescription	VarChar(256)	Y	Short description of the promotion offer
Description	Text	Y	Description of the promotion offer  Data entered here is not displayed on the Web store
Logo	Image	Y	Logo of promotion offer to display on the web store
AlternateLogolmageText	VarChar(512)	Y	Text to be displayed when the Logo image cannot be displayed.
MerchantID	Int	N	Merchant this promotion is defined for  FK reference to Merchants table
PromotionID	Int	N	FK reference to Promotions table
HeadLineDescription	Text	Y	Head line to display which describes the promotion
CodeEntryPageLinkText	Text	Y	Text for the link to the promotion code entry screen
SampleImageHelpText	VarChar(80)	Y	Sample text to display with the sample image
SampleImage	Image	Y	Sample image to display on the clickable link for this promotion. The image will most probably indicate where can the user find the PromotionCode on the coupon the user has
AlternateSampleImageText	VarChar(512)	Y	Text to be displayed when the Sample Image cannot be displayed.
FOPSetID	Int	Y	Forms of payments applicable to this promotion offer. The guest cannot select any form of payment other than the one defined under this fop set
PromotionOfferGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKPromotionOfferID	P	PromotionOfferID	Primary key.
IXPromoOfferMerchantPromoID		MerchantID, PromotionID	For query to get marketing messages for a given PromotionID & MerchantID

### 13.8 PromotionValidations

A PromotionValidations table holds all validations that must be met in order to use this promotion. This table is used to get all validations that need to be done to select a promotion. The data to validate can be defined as a range (Start and end range of zip codes) or as a single data validation (ValidationData).

Column	Type	Allow Nulls	Description
PromotionValidationID	Int	N	Primary key. Obtained from GatewayCounters.
Name	VarChar(80)	N	Name of the promotion validation
PromotionID	Int	N	FK reference to Promotions table (PromotionID column)
StartRange	VarChar(30)	Y	Start data value to validate if the validation is a range (i.e. Zip Code range)
EndRange	Varchar(30)	Y	End data value to validate if the validation is a range
ValidationData	VarChar(30)	Y	Data to validate if the validation is not a range (i.e. State)
ValidatePassPostalCode	Bit	Y	When set to 1, Web store enforces the promotion offer validation against the Postal Code entered for pass demographic
ValidateBillingPostalCode	Bit	Y	When set to 1, Web store enforces the promotion offer validation against the Postal Code entered for order's billing information
PromotionValidationGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKPromotionValidationID	P	PromotionValidationID	Primary key.
IXPrmValPrmCodeID	IX	PromotionCodeID	Used for query to get all promotion validations for a given promotion code

div class="pagebreak">

### 13.9 ResellerPriceSchedules

The ResellerPriceSchedules table defines Reseller Sales Channel PLU Detail values related to base PLU PriceSchedules.

#### Columns

Column	Type	Allow Nulls	Description
ResellerPriceScheduleID	Int	N	Unique Identifier
SalesChannelDetailID	Int	N	FK reference to SalesChannelDetails. SalesChannelDetailID
PriceScheduleID	Int	Y	FK reference to PriceSchedules.PriceScheduleID
ExternalPrice	Money	Y	Price charged by the Attraction to the Reseller
RetailPrice	Money	Y	Retail Price charged by the Reseller to the Customer
MinimumRetailPrice	Money	Y	Minimum retail price set by the Attraction
MaximumRetailPrice	Money	Y	Maximum retail price set by the Attraction
SPPriceScheduleID	Int	Y	FK reference to SPPPriceSchedules.SPPPriceScheduleID
PLU	NChar(20)	Y	Package Detail PLU for packages with 'Sum of Details' pricing method
ResellerPriceScheduleGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKResellerPriceScheduleID	P	ResellerPriceScheduleID	Primary Key
IXSalesChannelDetailPriceScheduleID	IX	SalesChannelDetailID,PriceScheduleID	

### 13.10 ResellerSalesChannelDetails

This table holds reseller-specific data associated to the sales channel.

#### Columns

Column	Type	Allow Nulls	Description
ResellerSalesChannelDetailID	Int	N	Unique Identifier
SalesChannelDetailID	Int	N	Foreign Key to SalesChannelDetail that this entry is associated to
SenderName	Text	Y	The name to show on sent email
ReplyTo	Text	Y	The reply to address for sent email
ReturnLocalStore	Bit	Y	Must tickets be returned to the local store
AllowTicketReturns	Bit	Y	Allows return of tickets sold previously
MaxDaysAfterSale	Int	Y	The number of days off sale the ticket can be returned
CategoryTemplate	Bit	Y	This is a category template
TemplateCategoryID	Int	Y	Points to the template
PasswordWebTemplateID	Int	Y	The template to use for the password email
TransReceipteGalaxyTemplateID	Int	Y	The web template to use for transaction receipts
ReturnReceipteGalaxyTemplateID	Int	Y	The web template to use for return receipts
URLWebTemplateID	Int	Y	The web template to user for the URL/UserID email
CategoryURL	Varchar(256)	Y	The URL for this category
RetailPrice	Money	Y	The retail price set by the site admin
ExternalPrice	Money	Y	The price charged by the venue
NetPrice	Money	Y	The price charged by the Reseller
MinimumRetailPrice	Money	Y	The lowest price the store can sell this item can be sold for
MaximumRetailPrice	Money	Y	The maximum price the store can sell this item can be sold for
ExternalPLU	Varchar(100)	Y	The item number set by the local administrator
DisplayItem	Bit	Y	Display Item flag
FirstDayOfSale	DateTime	Y	The first day this item can be sold
LastDayOfSale	DateTime	Y	The last day this item can be sold
AgencyID	Int	Y	Pointer to the Agency for this Category
FOPGroupID	Int	Y	The FOP Group for this sales channel
CodeTableID	Int	Y	Reference to the CodeTable for Security questions
SenderAddress	Text	Y	The email senders address
StoreOffline	Bit	Y	Flag for turning store off
TimeZoneID	VarChar(64)	Y	ID of time zone
CustomerID	Integer	Y	Foreign key to Customers.CustomerID
PrepayReceipteGalaxyTemplateID	Int	Y	The PDF template to use for prepayment receipts. Foreign key to the eGalaxyTemplates table
DeliveryMethodID	Int	Y	The optional Delivery Method used to send an SMS notification when a Transaction from Reseller is processed using this Sales Channel. Only Mobile-type Delivery Methods are valid here.
PriceAdjustmentType	Int	Y	The type of the price adjustment <sup>1</sup>
PriceAdjustmentValue	Money	Y	Percentage or dollar amount used to calculate the retail price on top of the Reseller price
BookingReferenceCollectionType	TinyInt	Y	Configuration option to control if the Booking Reference field is required <sup>2</sup>
TravelDateCollectionType	TinyInt	Y	Configuration option to control if the Travel Date field is required <sup>2</sup>
ResellerSalesChannelDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKResellerSalesChannelDetailID	P	ResellerSalesChannelDetailID	Primary key

#### <sup>1</sup> PriceAdjustmentType Values

Value	Gateway Constant Name	Description
0		Normal
1		Amount
2		Percentage

#### <sup>2</sup> BookingReferenceRequirement and TravelDateRequirement Values

Value	Gateway Constant Name	Description
0	reqDoNotCollect	Do not collect – Webstore will not present fields to collect the data
1	reqOptionalCollect	Collect (optional) – Webstore will present the fields to collect data, but they will not be required for the CreateTransaction message to successfully complete the transaction
2	reqRequiredCheck	Collect (Required) - Webstore will present the fields to collect data, and the data will be required for the CreateTransaction message to successfully complete the transaction

### 13.11 SalesChannelOptions

The SalesChannelOptions table stores Name/Value pairs associated to a SalesChannel or a Sales Channel Detail.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelOptionID	int	N	Primary key, always unique, identity
SalesChannelID	int	N	FK to SalesChannels.SalesChannelID
SalesChannelDetailID	int	Y	FK to SalesChannelDetails.SalesChannelDetailID
Name	varchar(50)	Y	Name of the option
Value	nvarchar(max)	N	Value of the option
ResellerSalesChannelDetailID	int	Y	FK to ResellerSalesChannelDetails.ResellerSalesChannelDetailID
DataType	int	Y	Type of the configuration option value <sup>1</sup>
Code	int	Y	Constant associated with a particular Sales Channel configuration setting <sup>2</sup>
SalesChannelOptionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelOptionID	P	SalesChannelOptionID	Primary Key
IXSalesChannelOptionCode		Code	Reduce query time

#### <sup>1</sup> DataType Values

Value	Gateway Constant Name	Description
1	INTEGER_DATA_TYPE	Value is an Integer
2	STRING_DATA_TYPE	Value is an String
3	BOOLEAN_DATA_TYPE	Value is a Boolean (Stored as YES/NO)
4	DATE_TIME_DATA_TYPE	Value is a DateTime (Stored as YYYYMMDD HH:MM using military time)
5	DATE_DATA_TYPE	Value is a Date (Stored as YYYYMMDD)
6	TIME_DATA_TYPE	Value is a Time (Stored as HH:MM using military time)
7	CURRENCY_DATA_TYPE	Value is a Currency
8	REAL_DATA_TYPE	Value is a Real/Double

#### <sup>2</sup> Code Values

Value	Gateway Constant Name	Description
0	SC_CODE_INVALID	[ Invalid Code ]
1	SC_GALAXY_CONNECT_SALES_CHANNEL_DETAIL	GalaxyConnect
2	SC_GALAXY_CONNECT_REQUIRES_OFFLINE_DELIVERY	RequiresOfflineDelivery
3	SC_GALAXY_CONNECT_APP_CONFIG_ID	GalaxyConnectAppConfigID
4	SC_GALAXY_CONNECT_CUSTOMER_ID	customerID
5	SC_GALAXY_CONNECT_SALES_PROGRAM_ID	salesProgramID
6	SC_GALAXY_CONNECT_CHANNEL_STATUS	ChannelStatus
7	SC_HIDE_UNAVAILABLE_PRODUCTS_FOR_SELECTED_EVENT	Hide unavailable products for selected event
8	SC_DEFAULT_MERCHANT_ID	Sales channel default merchant ID
9	SC_DETAIL_SHOW_EVENT_CAPACITY	Show event capacity
10	SC_DETAIL_ALLOW_MULT_EVENT_SELECT	Allow multiple event selection
11	SC_DETAIL_EGALAXY_SOURCE_ID	eGalaxy SourceID

### 13.12 SalesChannelPublishCommands

The SalesChannelPublishCommands table stores the publish requests created by Galaxy and MWS users to publish sales channel related data to the eGalaxy Webstore and kiosk.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelPublishCommandID	Integer	N	Primary key. Obtained from GatewayCounters.
SalesChannelID	Integer	N	Sales Channel being published
ExternalConnectionID	Integer	N	Foreign Key to ExternalConnections
PublishKind	Integer	N	Kind of publish request <sup>1</sup>
EffectiveDate	DateTime	Y	If assigned, publish command will not execute before the specified date
Email	nvarchar(128)	Y	Email address that will be notified of publishing results
UserID	Integer	N	Galaxy User that initiated the publish command
PerformCentralPublish	Bit	Y	1/true if the Web Publishing service should perform a central database publish operation before performing this web publish operation.

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelPublishCommands	P	SalesChannelPublishCommandID	Primary key

#### <sup>1</sup> PublishKind Values

Value	Gateway Constant Name	Description
0	pkKiosk	Kiosk publish
1	pkWebStore	Web store publish

### 13.13 SalesChannelPublishCommandDetails

The SalesChannelPublishCommandDetails table stores the associated information for publish requests in the SalesChannelPublishCommands table.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelPublishCmdDtID	Integer	N	Primary key. Obtained from GatewayCounters.
SalesChannelPublishCommandID	Integer	N	Foreign Key to SalesChannelPublishCommands
Key	nvarchar(128)	N	Detail Key, used to identify individual publish options within a command
Value	nvarchar(max)	Y	Value for Detail

#### Indexes

Name	Kind	Columns	Purpose
PKSCPCDetails	P	SalesChannelPublishCmdDtID	Primary key
IXSCPCommandID		SalesChannelPublishCommandID	

### 13.14 SalesChannels

The SalesChannels table contains definition of a sales channel. A Sales Channel is a location where tickets are sold. This table is used the web store and kiosk.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelID	Int	N	Primary key. Obtained from GatewayCounters.
Name	VarChar(80)	N	Name of sales channel.
ExternalID	VarChar(30)	Y	Used to identify the sales channel
Description	Text	Y	Long description of the sales channel.
URL	VarChar(256)	Y	URL of the web store where all details of this sales channel are published
MaxCreditCardUse	Int	Y	Maximum number of time the same credit card can be used (for the days defined under the CreditCardUseFrequency option) to purchase tickets  Zero = Unlimited  Non zero = Maximum credit card uses
CreditCardUseFrequency	Int	Y	Frequency in number of days, a credit card is allowed to be used  Zero = Unlimited  Non zero = Frequency of use of credit card use
MaxQtyLimit	Int	N	Maximum number of quantity allowed in a single internet transaction  Zero = Unlimited  Non zero = Maximum number of quantity.
ExternalConnectionID	Int	Y	FK reference to ExternalConnections table
MaxTicketQuantityEnforcement	Int	Y	Determines how to enforce the maximum ticket quantity
IgnoreFees	Bit	Y	True - Fees are never applied  False - Fees are applied based on other options
ItemizeFees	Bit	Y	True - Show individual Fee PLU  False - Combine fees by Fee PLU
HideFees	Bit	Y	True - Do not show fees on Web Store until checkout  False - Add fees to transaction as they are incurred
ApplyTransactionFee	Bit	Y	True - Apply Transaction fee  False - Do not apply Transaction fee
TransactionFee	Varchar(20)	Y	PLU selected as the transaction fee
ApplyItemFee	Bit	Y	True - Apply per Item fee  False - Do not apply per Item fee
ItemFee	Varchar(20)	Y	PLU selected as the item fee
SiteID	Varchar(128)	Y	A SiteID represents the unique identifier for the Sales Channel at an attraction. Any alphanumeric value can be entered in this field.
DisableUpsellReplacements	Bit	Y	Indicates if upsell replacements are disabled for this sales channel and all of its details.
DisableUpsellAddOns	Bit	Y	Indicates if upsell add-ons are disabled for this sales channel and all of its details.
SalesChannelGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelID	P	SalesChannelID	Primary key.

### 13.15 SalesChannelDeliveryMethods

The SalesChannelDeliveryMethods table will define the delivery methods that are valid for a given Sales Channel Category Group, as well as the ordering that will be used for the display of those delivery methods.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelDeliveryMethodID	Int	N	Primary key, always unique
SalesChannelDetailID	Int	N	Foreign Key to SalesChannelDetails.SalesChannelDetailID
DeliveryMethodID	Int	N	Foreign Key to DeliveryMethods.DeliveryMethodID
Sequence	Int	Y	Sequence number to control the order in which the delivery methods for the Sales Channel Category Group are displayed
SalesChannelDeliveryMethodGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelDeliveryMethodID	P	SalesChannelDeliveryMethodID	Primary Key.
IXSalesChannelDetailID	F	SalesChannelDetailID	Foreign Key to SalesChannelDetails.SalesChannelDetailID

### 13.16 SalesChannelDetails

The SalesChannelDetails table contains details of a sales channel. Different types of Sales Channel details include Sales Category Group, Sales Category, Sales Sub Category, Sales Sub Category Detail, and Promotion Offer. Different attributes of each detail can be found in the same table. In a row, the column DetailType identifies what type of detail is in the row (Sales Category Group or Sales Category etc.). The value in the ParentID is used to find the parent of this detail. This table is used the web store and kiosk.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelDetailID	Int	No	Primary key. Obtained from GatewayCounters.
SalesChannelID	Int	No	FK reference to SalesChannels table
Name	VarChar(256)	No	Name of the detail
ExternalID	VarChar(50)	Yes	Used to identify the sales channel detail. The URL query string of the web store uses data in this field
ShortName	VarChar(256)	Yes	Short name of a detail
Description	Text	Yes	Description about the detail
LongDescription	Text	Yes	Long description about the detail
HelpText	Text	Yes	Help text
DetailType	Int	No	Type of detail this row represents <sup>1</sup>
ParentID	Int	Yes	FK reference to the unique ID of the table referenced by the detail
Sequence	Int	No	Sequence number to use when this detail is displayed on web store or kiosk
ImageData	Image	Yes	Image or logo to display with the detail
AlternateImageText	VarChar(512)	Yes	Color of the buttons used to display the Sales Category. The color can be stored as HTML code
ButtonColor	VarChar(128)	Yes	Type of detail this row represents <sup>1</sup>
TextOnCalendar	VarChar(256)	Yes	Text to display on the Calendar shown when a guest selects this detail and if event tickets are present under this detail
EffectiveDate	DateTime	Yes	Date and time this sales category group needs to be effective
Identifier	VarChar(24)	Yes	Field to identify where this sales category group belongs. This field is used to identify the sales category group based on the Park selected by the guest
IdentifierHeaderText	VarChar(256)	Yes	Header text to display on the confirmation e-mail if tickets from the given sales category group are purchased by the guest  Data in this column is used by keyword @PARK_HEADER_TEXT
IdentifierLogo	VarChar(256)	Yes	Image to display on the confirmation e-mail if tickets from the given sales category group are purchased by the guest  Data in this column is used by keyword @PARK_LOGO
MerchantID	Int	Yes	FK reference to the Merchants table
SalesProgramID	Int	Yes	FK reference to SalesPrograms table  The Sales Program defined here has the list of items (item groups/individual PLUs) that can be displayed on the web store or kiosk. Rate is also defined on this sales program which is applied to the items selected in this sales program
CCMClientID	Int	Yes	Merchant's ID to use for the credit card processing
CCMUserName	VarChar(128)	Yes	Merchant's name to use for the credit card processing
CCMPassword	VarChar(128)	Yes	Password to use when authorizing credit card transactions on the Web store
SurveyID	Int	Yes	FK reference to the Surveys table
MarketingMessageID	Int	Yes	FK reference to MarketingMessages table  The message is displayed on the confirmation page and e-mail when the guest selects this detail
PromotionOfferID	Int	Yes	FK reference to PromotionOffers table
PLU	Char(20)	Yes	Individual PLU that is part of the Sales Sub Category defined in this row  FK reference to Items (PLU column) table
MaxTktQtyAllowedInTran	Int	Yes	Maximum number of tickets a guest can purchase under this detail in single transaction. Applies to all individual tickets defined under this detail. If 0 or NULL, no restrictions
OwnerType	Int	Yes	Specifies who owns this detail (mostly used for Sales Category Group detail - DetailType of 0). The values are defined in <i>Owner Type Values table below</i> <sup>2</sup>
OwnerID	Int	Yes	ID of the owner of this detail (mostly used for Sales Category Group detail - DetailType of 0)
URL	VarChar(256)	Yes	URL of the page to show when this detail is selected (or clicked) by the guest.  This is used for Promotion Offers, when user selects a promotion offer the page on the URL entered here is displayed
WebPageTemplate	VarChar(80)	Yes	FK reference to WebTemplates table. Used for assigning Marketing Message templates on the SalesChannelGroup detail
WebPageHeaderTemplate	VarChar(80)	Yes	Name of the header web page template to use for all web pages displayed after this merchant is selected  This applies to Sales Category Group detail only
WebPageFooterTemplate	VarChar(80)	Yes	Name of the footer web page template to use for all web pages displayed after this merchant is selected  This applies to Sales Category Group detail only
WebPageStyleSheet	VarChar(80)	Yes	Name of the web page style sheet to use for all web pages displayed after this merchant is selected.  A Stylesheet can be used for font styles, colors, sizes, background images or background colors of the web page  This applies to Sales Category Group detail only
DisplaySetID	Int	Yes	
UpsellID	Int	Yes	FK reference to Upsell table
MarkAsDefault	Bit	Yes	Set to 1 to mark a SalesCategoryGroup detail to be used as a default when no merchant is selected on the web store
EventTypeID	Int	Yes	FK reference to RMEventTypes table
AlternateLogoText	VarChar(512)	Yes	Text to be displayed when the Attraction Identifier logo cannot be displayed.
IsPassRequired	Bit	Yes	Set to 1 to enable a SalesCategory or SalesSubCategory detail for PassRequired tickets. When this flag is set to 1, tickets under

			this detail must be Pass Required tickets.
EnablePaymentPlans	Bit	Yes	When set to 1, the payment plan is enabled for this detail. This option is only available for Category and Sub-Category detail.
CalendarDisplay	Integer	Yes	A number indicating the status of the calendar display on the webstore. The values are defined in <i>Calendar Status Values table below</i> <sup>3</sup> .
EnableGroupSales	Bit	Yes	Set to 1 to enable Group Sales on the eGalaxy Web Store
AllowDiscount	Bit	Yes	Set to 1 to allow a discount
ResellerWebStore	Bit	Yes	Is this a Reseller Web Store?
MaxTicketQuantityEnforcement	Int	Yes	Enforce Max Ticket Quantity
ItemizeFees	Bit	Yes	True - Show individual Fee PLU False - Combine fees by Fee PLU
HideFees	Bit	Yes	True - Do not show fees on Web Store until checkout False - Add fees to transaction as they are incurred
ApplyTransactionFee	Bit	Yes	True - Apply Transaction fee False - Do not apply Transaction fee
TransactionFee	Varchar(20)	Yes	PLU selected as the transaction fee
ApplyItemFee	Bit	Yes	True - Apply per Item fee False - Do not apply per Item fee
ItemFee	Varchar(20)	Yes	PLU selected as the item fee
TransportationType	Integer	Yes	Used for SalesChannel Categories to hold the transportation type to be used for the category. The values are defined in <i>Transportation Type Values table below</i> <sup>4</sup>
RequireCustomerLogin	Bit	Yes	When set to true, this will inform the eGalaxy Web Store that it should require the user to login to the Category Group before they are able to select items from the Navigation Panel or navigate to the View Items page.
DisplayNavLink	Bit	Yes	When set to true, this will inform the eGalaxy Web Store that a "Navigation Display Link" section should be added above the existing Navigation Panel. Any Category Group that has this flag activated will automatically be listed as link in the Navigation Link Panel.
CategoryType <sup>5</sup>	Int	Yes	Defines the context of the category
RequireGuestNames	Bit	Yes	Requires the reseller webstore to include guest names for each ticket for this sales channel.
RestrictByIPAddress	Bit	Yes	Allow Reseller web store to restrict access by IP Address
ShowSoldOutEvents	Bit	Yes	If enabled, all events for the day are displayed, including sold out events.
AutoSelectEvents	Bit	Yes	If only one event is available, the system will automatically select this event without prompting the user.
ShareEventCalendars	Bit	Yes	If enabled, all events on this Category or Sub Category that are of the same event type will share a common calendar.
MinTktQtyAllowedInTran	Integer	Yes	Minimum number of tickets a guest can purchase under this detail in single transaction. Applies to all individual tickets defined under this detail. If 0 or NULL, no restrictions.
CustomerFieldAttributeGroupID	Integer	Yes	Attach required group sales fields to sales category
UpsellReplacementScriptTemplateID	Integer	Yes	Foreign Key reference to WebTemplates table for records that have a type of TEMPLATE_KIND_UPSELL_MESSAGE (5). This script will be used to generate the message to display upsell options to the user for Replacement Upsell items.
UpsellAddOnScriptTemplateID	Integer	Yes	Foreign Key reference to WebTemplates table for records that have a type of TEMPLATE_KIND_UPSELL_MESSAGE (5). This script will be used to generate the message to display upsell options to the user for Add On Upsell items.
UpsellReplacementHeaderScriptTemplateID	Integer	Yes	Foreign key reference to the WebTemplates table for records that have a type of TEMPLATE_KIND_UPSELL_MESSAGE (5). This template will be used to generate the header message for upsell add-ons.
UpsellAddOnHeaderScriptTemplateID	Integer	Yes	Foreign key reference to the WebTemplates table for records that have a type of TEMPLATE_KIND_UPSELL_MESSAGE (5). This template will be used to generate the header message for upsell replacements.
DisableUpsellReplacements	Bit	Yes	Indicates if upsell replacements are disabled for this sales channel detail, and all sub-details.
DisableUpsellAddOns	Bit	Yes	Indicates if upsell add-ons are disabled for this sales channel detail, and all sub-details.
ShowCapacity	Bit	Yes	If enabled, the web store will display the remaining capacity for events, and allow the guest to select multiple event times at once within a single day. Note that the use of this feature disables the ability to specify the quantity of tickets desired prior to selecting the calendar. It also disallows sharing of event calendars.
DonationPLU	Varchar(20)	Yes	Specified the donation PLU to be used
DonationPromptType	Int	Yes	Specifies the donation prompt type (values TBD)
AllowPriceOverride	Bit	Yes	This column determines if the price of an item that is not a donation or stored value can be overridden by the guest.
HidePackageDetails	Bit	Yes	Control if the detail elements of packages will be displayed on the mobile webstore.
CalculateResellerPrice	Bit	Yes	Defines whether or not the Category Group, Category and PLU Detail automatically calculates the Reseller price based on the default sales program of the customer defined on the reseller category.
AssignPrimaryPass	Bit	Yes	AssignPrimaryPass allows a Category Group to specify that passes on a given order should be linked through the Passes.Master field. When AssignPrimaryPass is enabled, the eGalaxy Webstore will specify which pass is the master.
DoNotStoreEndorsement	Bit	Yes	If the bit is set, the consumer of the SalesChannelDetail data (typically the eGalaxy Webstore) will not record (encrypted or not) the PAN (payment account number).
BillingFieldAttributeGroupID	Integer	Yes	This column is a foreign key reference to the FieldAttributeGroups table which will be used to determine the billings fields.
DeliveryMethodGroupID	Integer	Yes	FK to DeliveryMethodGroups. Will allow the specification of the DeliveryMethodGroup at the Category Group, Category, Sub-Category and PLU detail levels rather than needing to select it for each item within the Sales Channel.
GiftedPurchase	Bit	Yes	Allow the Category Group to be used to purchase tickets that will be sent to the ship to contact as a gift, while the bill to contact will receive the payment receipt.
CommunicationGroupID	UniqueIdentifier	Y	A unique identifier that is used to associate records in the OrderCommunications table with a customer, customer category, or sales channel category group.
UpsellOptionsToDisplay	Int	Yes	Number of upsell options to display for a given product by default.
EnableGiftAid	Bit	Yes	True if GiftAid is enabled on the web store for this category group, false otherwise.
PostalCodeRangeID	Integer	Yes	FK to PostalCodeRanges. Will allow the specification of the PostalCodeRange at the Category Group, Category, Sub-Category and PLU detail levels, for use with Postal Code Range Filtering during Kiosk sales.
ActivatedPassStatus	Integer	Yes	New pass status for activated passes.
EnableResellerPriceSchedules	Bit	Yes	Indicates if the Reseller Price Schedules functionality is enabled for this sales channel detail, and all sub-details.
EnableGiftAid	Bit	Yes	Indicates if Gift Aid functionality is enabled for this web store category group.
CollectVisitDate	Integer	Yes	Determines whether a category group or a category should prompt the guest for their visit date. This setting is only valid at the

			category group or category level. <sup>6</sup>
ShowPhotoOnMemberPortal	Bit	Yes	Controls whether or not guest photos are displayed on the Online Member Portal.
ShowPhotosPendingApproval	Bit	Yes	Controls whether or not guest photos pending approval are displayed on the Online Member Portal.
AllowPhotoUploadOnMemberPortal	Bit	Yes	Controls whether or not guest photo uploads are allowed in the Online Member Portal.
AllowPhotoReplacementOnMemberPortal	Bit	Yes	Controls whether or not guest photos can be replaced in the Online Member Portal.
AllowEntitlementOddOn	Integer	Yes	Controls whether or not Entitlement Add On are allowed in the Online Member Portal. <sup>7</sup>
DefaultDeliveryMethodID	Integer	Yes	Defines the default value for the Delivery method for Category Group in the Online Member Portal.
AutoSelectUpsellOptionsByDate	Bit	Yes	Enables filtering on upsell addons by the source product date (visit date or ticket date).
SalesChannelDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**NOTES:**

- Data entered in the fields Name, ShortName, ButtonColor, TextOnCalendar, and IdentifierHeaderText may not be displayed properly on the web store. The size of the field is much larger than what can be displayed on the web store. This is because the user can enter the HTML code within these fields to indicate the font color or font style etc. within the same field
- Fields Description, Long Description, and HelpText have a data type of Text and therefore these fields can have HTML code and images in them.

**Indexes**

Name	Kind	Columns	Purpose
PKSalesChannelDetailID	P	SalesChannelDetailID	Primary key.
IXSalCatGrpsSalesChnlID		SalesChannelID	For query to get all sales category groups for a sales channel
IXSalesChannelIDParentID	IX	SalesChannelID, ParentID	For query to load all sales channel details

**<sup>1</sup> DetailType Values**

Value	Gateway Constant Name	Description
0	ID_SALES_CAT_GROUP_DETAIL	Sales Category Group, detail type. ParentID is the FK reference to the SalesChannels table
1	ID_SALES_CAT_DETAIL	Sales Category, detail type. ParentID is the Sales Category Group this Sales Category belongs to
2	ID_SALES_SUB_CAT_DETAIL	Sales Sub Category, detail type. ParentID is the Sales Category this Sales Sub Category belongs to
3	ID_SALES_SUB_CAT_DETAIL_DETAIL	Sales Sub Category Detail (individual PLU), detail type. ParentID is the Sales Sub Category this PLU belongs to
4	ID_PROMOTION_OFFER_DETAIL	Promotion Offer. ParentID is the Sales Category Group, Sales Category, or Sales Sub Category

**<sup>2</sup> OwnerType Values**

Value	Gateway Constant Name	Description
0	ID_MERCHANT_OWNER	Value in the OwnderID column is FK reference to the Merchants table. The merchant owns this detail
1	ID_PROMOTION_OWNER	Value in the OwnderID column is FK reference to the Promotions table. The promotion owns this detail

**<sup>3</sup> Calendar Status Values**

Value	Gateway Constant Name	Description
0	CALENDAR_DISPLAY_DATE_TIME	Display the date selection first and then the available event times.
1	CALENDAR_DISPLAY_DATE	Only display the date selection. Do not display the time selection.
2	CALENDAR_DISPLAY_EVENT	No calendar to be displayed. No date and time picker. The event is the first thing displayed.

**<sup>4</sup> TransportationType Values**

Value	Gateway Constant Name	Description
0	TRANSPORTATION_TYPE_NONE	None
1	TRANSPORTATION_TYPE_SCHEDULE	Schedule transportation type
2	TRANSPORTATION_TYPE_PURCHASE	Purchase transportation type
3	TRANSPORTATION_TYPE_COMMUTER	Commuter transportation type

**<sup>5</sup> CategoryType Values**

Value	Gateway Constant Name	Description
0	ctDefault	The default behavior for a Sales Channel Category
1	ctReseller	Defines the Sales Channel Category as a Reseller web store
2	ctRenewal	Defines the Sales Channel Category as a Pass Renewals for the web store
3	ctGroupSales	Enables a Sales Channel Category Group for the GroupSales on the Web module
4	ctMobileSales	Enables a Sales Channel Category Group for the Mobile on the Web module
5	ctActivation	Defines the Sales Channel Category for Pass Activation
6	ctReservedSeat	Defines the Sales Channel Category for Reserved Seat
7	ctOnlineMemberPortal	Enables a Sales Channel Category Group for the Online Member Portal Web module
8	ctMemberDeals	Defines the Sales Channel Category for Member Portal Deals
9	ctMemberPurchases	Defines the Sales Channel Category for Member Purchases
10	ctStandard	The standard behavior for a Sales Channel Category Group
11	ctMemberBenefits	Defines the sales channel category for Member Benefits

**<sup>6</sup> CollectVisitDate Values**

Value	Gateway Constant Name	Description
0	TCollectVisitDateType.gvtNotSet	Indicates this option has not been set or is not valid at this level (eg., at the subcategory or PLU level)
1	TCollectVisitDateType.gvtCollectVisitDate	Prompt for visit date

2	TCollectVisitDateType.gvtDoNotCollectVisitDate	Do not prompt for visit date
3	TCollectVisitDateType.gvtUseParentSetting	Use setting at parent level (only valid for category; causes category to use the setting at the category group level)

#### **7 AllowEntitlementAddOnType Values**

Value	Gateway Constant Name	Description
0	TAAllowEntitlementAddOnType.eaotUnknown	Indicates this option has not been set or is not valid at this level (eg., at the category, subcategory or PLU level)
1	TAAllowEntitlementAddOnType.eaotAllow	Allow Entitlement Add On at this level
2	TAAllowEntitlementAddOnType.eaotDisallow	Do not allow Entitlement Add On at this level

### 13.17 SalesChannelIPAddresses

The SalesChannelIPAddresses table connects sales channel detail with IP Address Group(s).

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelIPAddressID	Int	No	Primary key. Obtained from GatewayCounters.
SalesChannelDetailID	Int	No	Foreign key into SalesChannelDetails table
IPAddressGroupID	Int	No	Foreign key into IPAddressGroups table
SalesChannelIPAddressGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PK_SalesChannelIPAddressID	P	SalesChannelIPAddressID	Primary key.

### 13.18 SalesChannelLanguages

Defines which translation languages are supported by a given sales channel.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelLanguageID	int	N	Primary key, always unique
SalesChannelDetailID	int	N	FK to SalesChannelDetails.SalesChannelDetailID
LanguageID	int	N	FK to TranslationLanguages.TranslationLanguageID
SalesChannelLanguageGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelLanguageID	P	SalesChannelLanguageID	Primary Key

### 13.19 SalesChannelLoyaltyPoints

The SalesChannelLoyaltyPoints records have a 1-to-1 association with a sales channel category group, and store the number of bonus points awarded to an order if a ticket/item from that category group is purchased on the eGalaxy Web Store.

#### Columns

Column	Type	Allow Nulls	Description
SalesChannelLoyaltyPointID	Integer	N	Primary Key, always unique
SalesChannelDetailID	Integer	Y	Foreign Key to SalesChannelDetail table
LoyaltyPoints	Float	Y	The number of bonus points awarded to an order if a ticket or item from the category group is purchased on the eGalaxy Web Store.

#### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelNodeID	P	SalesChannelNodeID	Primary key.

## 13.20 SalesChannelNodes

The SalesChannelNodes table contains the nodes or agencies that are part of the Sales Channel. The column DetailType is identifies what type of detail is included in the Detail column.

### Columns

Column	Type	Allow Nulls	Description
SalesChannelNodeID	Int	N	Primary key. Obtained from GatewayCounters.
SalesChannelID	Int	N	FK reference to the SalesChannels table
DetailType	Int	N	Type of detail this row represents <sup>1</sup>
Detail	Int	N	Agency Id or the Node number. Agency or Node number defined here belongs to the given Sales Channel

### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelNodeID	P	SalesChannelNodeID	Primary key.
IXSalChnlNodesSalesChannelID		SalesChannelID	Used for query to get all nodes/agencies for a given SalesChannelID

### <sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
0	ID_AGENCY_ID_DETAIL	Agency detail type, value in the Detail column is the Agency ID
1	ID_NODE_NUMBER_DETAIL	Node number detail type, value in the Detail column is the Node number

## 13.21 SalesChannelPublishLog

This table stores the last time a publish was performed for a given table and Sales Channel.

### Columns

Column	Type	Allow Nulls	Description
SalesChannelPublishLogID	Int	N	Unique Identifier
SalesChannelID	Int	N	Foreign key to SalesChannels.SalesChannelID
TableID	Int	N	Constant value defining the table that was published <sup>1</sup>
PublishDateTime	DateTime	N	Date/Time of the last successful publish
AuxID	Int	N	Foreign key to UniqueID value for table specified by TableID

### Indexes

Name	Kind	Columns	Purpose
PKSalesChannelPublishLogID	P	SalesChannelPublishLogID	Primary Key
IXSCPLogTableIDSalesChannelID		TableID, SalesChannelID	For queries during a publish

<sup>1</sup>TableID Values

See Table ID information as Indexed at the beginning of this document.

## 13.22 SystemLogons

Stores the authentication information for a logon required to access certain areas of Gateway applications. The table contains a pair of Username and Password for each logon. Fields OwnerType and OwnerID identify who owns the logon.

### Columns

Column	Type	Allow Nulls	Description
SystemLogOnID	Int	N	Primary key, always unique
Username	Varchar(100)	N	User name for this login
Password	Varchar(128)	N	Password associated with the given user name.  Note: The password is encrypted
OwnerType	Int	N	Identifies the contents of the OwnerID column. <sup>1</sup>
OwnerId	Int	N	A numeric code uniquely identifying the owner of this record. Most likely the value in this column would be a foreign key to another table in Gateway's SQL database
RequirePasswordChange	Bit	Y	When set to 1, indicates that the user must change their password when they next logon
SaltValue	Char(10)	Y	Salt value used for one-way password hashing
PasswordEncryptionMethod	Int	Y	Encryption method used for password where 0/null = legacy encryption and 1 = one-way hash.

### Indexes

Name	Kind	Columns	Purpose
PKSystemLogOnID	P	SystemLogOnID	Primary Key.
IXSystemLogOnsOwnerTypeOwnerId		OwnerType, OwnerID	Used by the query to get a system logon record by owner
IXSystemLogonsUsernamePassword		Username, Password	Used by query used when authenticating a logon

<sup>1</sup> OwnerType Values

Value	Gateway Constant Name	Description
1	OWNER_TYPE_CONTACT	Logon information for a Galaxy Contact. The value of <i>OwnerId</i> is a foreign key to <i>CustContacts.CustContactID</i>
2	OWNER_TYPE_CUSTOMER	Logon information for a Galaxy Customer. The value of <i>OwnerId</i> is a foreign key to <i>Customers.CustomerID</i>

### 13.23 TranslationEventNames

Stores the unique event names from the Resource Management module so that the translations can be performed once per unique event name instead of once per event.

#### Columns

Column	Type	Allow Nulls	Description
TranslationEventNameID	int	N	Primary key, always unique
EventName	varchar(100)	Y	Name of a resource managed event

#### Indexes

Name	Kind	Columns	Purpose
PKTranslationEventNameID	P	TranslationEventNameID	Primary Key
IXTranslationEventNamesEventName		EventName	

### 13.24 TranslationFields

Stores the list of fields that are translatable. This table is maintained internally via Galaxy, and should never be manually edited.

#### Columns

Column	Type	Allow Nulls	Description
TranslationFieldID	int	N	Primary key, always unique
TranslationTableID	int	Y	FK to TranslationTables.TranslationTableID
Name	nvarchar(50)	Y	Name of the field that has translatable content.
ForeignKeyFilter	nvarchar(100)	Y	Holds the value of the table name and column that describe the connection to another table. Example: WebTemplates_Upsell
TranslationFieldGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTranslationFieldID	P	TranslationFieldID	Primary Key
IXTranslationFieldTableID		TranslationTableID	
IXTranslationFieldsForeignKeyFilter		ForeignKeyFilter	

### 13.25 TranslationLanguages

The TranslationLanguages table stores all of the languages that are available for translation on the web store. The actual languages that are enabled per Sales Channel is controlled by the SalesChannelLanguages table.

#### Columns

Column	Type	Allow Nulls	Description
TranslationLanguageID	int	N	Primary key, always unique
Name	nvarchar(50)	Y	Name of the language
BaseLanguage	bit	Y	True if the language represents the base language of the installation - this is the language used by the text within the standard tables (e.g. Items). Only one record is allowed to have a value of 1 for this field, and this selection should not be changed once translation values are stored in the system.
FlagImage	image	Y	A image file (e.g. BMP) that graphically represents this language. Typically an image of a flag.
CultureName	nvarchar(10)	Y	The .NET culture name for this language. This is used by the webstore to correctly display numeric information such as currency and dates.
Sequence	int	Y	Used to define the order in which the languages will be displayed.
DateFormatType	Int	Y	Indicates which of the predefined date formats will be used when performing date-related translations. <sup>1</sup>
TimeFormatType	Int	Y	Indicates which of the predefined time formats will be used when performing date-related translations. <sup>2</sup>
TranslationLanguageGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTranslationLanguageID	P	TranslationLanguageID	Primary Key
IXTranslationLanguagesBaseLang		BaseLanguage	
IXTranslationLanguagesSequence		Sequence	

#### <sup>1</sup> DateFormatType Values

Value	Gateway Constant Name	Description
0	DATEFORMATTYPE_DEFAULT	Use the default .NET short date format for the given culture name for this language
1	DATEFORMATTYPE_FORMAT1	YYYY-MM-DD, e.g. 2003-11-09
2	DATEFORMATTYPE_FORMAT2	MM-DD-YYYY, e.g. 11-09-2003
3	DATEFORMATTYPE_FORMAT3	DD-MM-YYYY, e.g. 09-11-2003
4	DATEFORMATTYPE_FORMAT4	mmm dd-YYYY, e.g. Nov 09-2003
5	DATEFORMATTYPE_FORMAT5	DD MONTH-YYYY, e.g. 09 November-2003
6	DATEFORMATTYPE_FORMAT6	YYYYMMMD, e.g. 2003NOV09
7	DATEFORMATTYPE_FORMAT7	YYYYMONTHDD, e.g. 2003November09
8	DATEFORMATTYPE_FORMAT8	YYYY-MMM-DD, e.g. 2003-NOV-09
9	DATEFORMATTYPE_FORMAT9	YYYY-MONTH-DD, e.g. 2003-November-09
10	DATEFORMATTYPE_FORMAT10	Day, Month DD, YYYY, e.g. Sunday, November 9, 2003
11	DATEFORMATTYPE_FORMAT11	MONTH DD, YYYY, e.g. November 09, 2003
12	DATEFORMATTYPE_FORMAT12	MMM DD, YYYY, e.g. NOV 09, 2003
13	DATEFORMATTYPE_FORMAT13	MM/DD/YYYY, e.g. 11/09/2003
14	DATEFORMATTYPE_FORMAT14	DD/MM/YYYY, e.g. 09/11/2003
15	DATEFORMATTYPE_FORMAT15	MONTH/DD/YYYY, e.g. November/09/2003
16	DATEFORMATTYPE_FORMAT16	DD/MONTH/YYYY, e.g. 09/November/2009
17	DATEFORMATTYPE_FORMAT17	YYYY/MMM/DD, e.g. 2003/NOV/09
18	DATEFORMATTYPE_FORMAT18	YYYY/MONTH/DD, e.g. 2003/November/09
19	DATEFORMATTYPE_FORMAT19	dd/mmm/yyyy, e.g. 09/Nov/2009
20	DATEFORMATTYPE_FORMAT20	dd mmm-yyyy, e.g. 09 Nov-2009
21	DATEFORMATTYPE_DEFAULT_LONG	Use the default .NET long date format for the given culture name for this language

#### <sup>2</sup> TimeFormatType Values

Value	Gateway Constant Name	Description
0	TRANSLATIONTIMEFORMAT_DEFAULT	Use the default .NET short time format for the given culture name for this language
1	TRANSLATIONTIMEFORMAT_1	h:mm tt e.g. 8:45 AM, 8:45 PM
2	TRANSLATIONTIMEFORMAT_2	hh:mm tt e.g. 08:45 AM, 08:45 PM
3	TRANSLATIONTIMEFORMAT_3	H:mm e.g. 8:45, 20:45
4	TRANSLATIONTIMEFORMAT_4	HH:mm e.g. 08:45, 20:45

### 13.26 TranslationTables

Stores the list of tables that have translatable content. This table is maintained internally via Galaxy, and should never be manually edited.

#### Columns

Column	Type	Allow Nulls	Description
TranslationTableID	int	N	Primary key, always unique
Name	nvarchar(50)	Y	Name of the table that has translatable content.
AuxIDFieldName	nvarchar(50)	Y	Name of the primary key field within the translatable table.
NameFieldName	nvarchar(50)	Y	Name of the field which contains a string representation of a given record within the table (e.g. Items.PLU).
TranslationTableGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTranslationTableID	P	TranslationTableID	Primary Key
IXTranslationTablesName		Name	

### 13.27 TranslationValues

Stores the translated content for the non-base languages.

#### Columns

Column	Type	Allow Nulls	Description
TranslationValueID	int	N	Primary key, always unique
TranslationLanguageID	int	Y	Foreign Key to TranslationLanguages.TranslationLanguageID
TranslationTableID	int	Y	Foreign Key to TranslationTables.TranslationTableID
TranslationFieldID	int	Y	Foreign Key to TranslationFields.TranslationFieldID
AuxTableRecordID	int	Y	Foreign Key to the PK of the table referenced by TranslationTableID
Value	nvarchar(max)	Y	The translation value
TranslationValueGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTranslationValueID	P	TranslationValueID	Primary Key
IXTranslationValuesLookupA		TranslationLanguageID, TranslationFieldID, AuxTableRecordID	
IXTranslationValuesLookupB		TranslationLanguageID, TranslationTableID, TranslationFieldID, AuxTableRecordID	Unique Index
IXTranslationValuesFieldRecordLanguage		TranslationFieldID, AuxTableRecordID, TranslationLanguageID	
IXTranslationValuesRecordFieldLanguageValue		AuxTableRecordID, TranslationFieldID, TranslationLanguageID, TranslationValueID	

### 13.28 Upsell

This table stores Marketing information to be used for a given Sales Channel when it is being applied as Upsell.

#### Columns

Column	Type	Allow Nulls	Description
UpsellID	Int	N	Primary key, always unique
Header	Varchar(256)	Y	Marketing message header
Body	Varchar(256)	Y	Marketing message body

#### Indexes

Name	Kind	Columns	Purpose
PKUpsellID	P	UpsellID	Primary Key.

### 13.29 WSHTML

This table contains the HTML code for Header, sub header, footer, sub footer, etc for a page.

#### Columns

Column	Type	Allow Nulls	Description
WSHTMLID	Int	N	Primary key, always unique
Name	Varchar(128)	N	This is a composite description fo HTML kind, Sales Channel Detail Name, and Langauge. This is a system created field and user does not directly modify it.
Kind	Int	N	This field indicated what kind of HTML it is: header, sub-header, footer, etc.
HTML	nvarchar(max)	N	HTML code
ParseDynamicTags	Bit	Y	Used to Dynamically Parsing Pass Tags on WebStore
WSHTMLGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKWSHTMLID	P	WSHTMLID	Primary Key.

#### <sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	wskHeader	Page Header
2	wskSubHeader	Page sub-header
3	wskFooter	Page Footer
4	wskSubFooter	Page sub-footer
5	wskNavigation,	Navigation controls
6	wskError,	Error page
7	wskLandingPage,	Landing Page
8	wskReviewOrder,	Review Order page
9	wskOrderConfirmation	Order Confirmation page
10	wskTermsAndConditions	Terms and Conditions page
11	wskPassLookup	Pass Lookup page
12	wskPassRenewal	Pass Renewal page
13	wskPaymentPlansLogin	Payment Plans Login page
14	wskGroupSalesLogin	Group Sales Login page
15	wskGroupSalesVerifyLimits	Group Sales Verify Limits page
16	wskPayment	Payment page
17	wskStoreOffLine	Store Off Line
18	wskViewItems	View Items Tags
19	wskViewCart	View Cart Tags
20	wskBillingInfo	Billing Info Tags
21	wskShippingInfo	Shipping Info Tags
22	wskDeliveryInfo	Delivery Info Tags
23	wskSurveyLandingPage	Survey Landing Page
24	wskSurveyConclusionPage	Survey Conclusion Page
25	wskMobileAccountIndex	
26	wskMobileAccountOrder	
27	wskMobileAccountOrderHistory	
28	wskMobileAccountSignIn	
29	wskMobileAccountSignOut	
30	wskMobileCheckoutBilling	
31	wskMobileCheckoutBillingReview	
32	wskMobileCheckoutCart	
33	wskMobileCheckoutDiscount	
34	wskMobileCheckoutGuestNames	
35	wskMobileCheckoutLoyalty	
36	wskMobileCheckoutOptions	
37	wskMobileCheckoutOrderSummary	
38	wskMobileCheckoutPayment	
39	wskMobileCheckoutShipping	
40	WskMobileCheckoutShippingReview	
41	wskMobileErrorIndex	
42	wskMobileProductsCategories	
43	wskMobileProductsDetails	
44	wskMobileProductsIndex	
45	wskMobileSiteHeader	
46	wskMobileSiteFooter	
47	wskMobileSiteIndex	

48	wskMobileSiteNoJavaScript	
49	wskMobileVisitDateIndex	
50	wskPOSSalesScript	
51	wskGiftAidAffidavit	Affidavit prompt content for Gift Aid
52	wskMemberPortalInfoPage	Member Portal Info Page
53	wskLogonPage	Logon Page
54	wskHomePage	Home Page
55	wskViewDemographicsPage	View Demographics Page
56	wskEditDemographicsPage	Edit Demographics Page
57	wskAddMembersPage	Add Members Page
58	wskRenewMemberPage	Renew Member Page
59	wskBenefitsPage	Benefits Page
60	wskRedeemedBenefitsPage	Redeemed Benefits Page
61	wskMemberCardPage	Member Card Page
62	wskEntitlementAddOnPage	Entitlement Add-on Page
63	wskDownloadsPage	Downloads Page
64	wsLinkAccountPage	Link Account Page
65	wsCreateAccountPage	Create Account Page
66	wskHead	Bottom of Page Head
67	wskBodyBegin	Top of Page Body
68	wskBodyEnd	Bottom of Page Body
69	wskStoreUpdating	Store Updating

### 13.30 WSHtmlConnections

This table contains the interconnection information between WSHTML and Merchants / Sales Channel Details (Group). The each entry defines the owner of the connection (ConnectionType) and the ID of the owner (ConnectionID). This allows many-to-many relationships between WSHTML and Merchants / Sales Channel Details (Group)

#### Columns

Column	Type	Allow Nulls	Description
WSHTMLConnectionID	Int	N	Primary key, always unique
WSHTMLID	Int	N	WSHTML foreign key for this connection
ConnectionType	Int	N	This value indicates who the connection is for (Merchants or Sales Channel)
ConnectionID	Int	N	Foreign key ID to the owner of the connection. <sup>1</sup>  Merchants: Merchants.MerchantID  Sales Channel Group: SalesChannelDetails.SalesChannelDetailID
TranslationLanguageID	Int	Y	FK to TranslationLanguages.TranslationLanguageID
WsHTMLConnectionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKWSHTMLConnectionID	P	WSHTMLConnectionID	A non-clustered Primary Key. Always unique
UCXWSHTMLConnHTMLIDTypeConnIDTransLangID	UCX	ConnectionType, ConnectionID, WSHTMLID, TranslationLanguageID	This is a Unique Clustered Index. This index prevents user from creating duplicated HTML connection entries for the connection (owner).

<sup>1</sup> ConnectionType Values

Value	Gateway Constant Name	Description
0	wsctSalesChannelGroup	The connection is for Sales Channel Details (Category Groups).
1	wsctMerchant	The connection is for Merchant
2	wsctSalesChannelCategory	The connection is for Sales Channel Details (Categories)

### 13.31 WebTemplates

The WebTemplates table houses both confirmation e-mail templates and marketing messages. Confirmation e-mail templates are attached to delivery methods and define the format of the confirmation e-mail sent when an order is processed via eGalaxy. Marketing messages appear on both the web store's confirmation page, as well as in the confirmation e-mail (if included in the confirmation e-mail template via GatewayScript).

Web templates can have both an HTML and Plain Text version specified. GatewayScript is supported for both web template kinds. Both the HTML and plain text bodies are parsed when generating confirmation e-mails via eGalaxy.

#### Columns

Column	Type	Allow Nulls	Description
WebTemplateID	Int	N	Primary key, always unique
TemplateName	Char(100)	N	Name of the web template
TemplateKind	Int	N	Kind of web template <sup>1</sup>
HTMLBody	nvarchar(100)	Y	HTML version of the template
PlainTextBody	nvarchar(100)	Y	Plain text version of the template
WebTemplateGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKWebTemplatesWebTemplateID	P	WebTemplateID	Primary Key

#### <sup>1</sup> TemplateKind Values

Value	Gateway Constant Name	Description
0	TEMPLATE_KIND_CONFIRMATION_EMAIL	Confirmation e-mail template
1	TEMPLATE_KIND_MARKETING_MESSAGE	Marketing message template
2	TEMPLATE_KIND_MOBILE_DELIVERY	Mobile delivery template
3	TEMPLATE_KIND_CONTRACT_STATEMENT	Contract statement template
4	TEMPLATE_KIND_USER_LOGON_EMAIL	User logon email template
5	TEMPLATE_KIND_UPSELL_MESSAGE	Upsell message template
6	TEMPLATE_KIND_GUEST_VISIT_SURVEY	Guest Visit Survey message template
7	TEMPLATE_KIND_HTML_OE_STATEMENT	Order Entry HTML statement template
8	TEMPLATE_KIND_CSV_OE_STATEMENT	Order Entry CSV statement template
9	TEMPLATE_KIND_PDF_OE_STATEMENT	Order Entry PDF statement template
10	TEMPLATE_KIND_HTML_INVOICE_CREDITMEMO	Invoice HTML template
11	TEMPLATE_KIND_GIFT_AID	Gift Aid message template
12	TEMPLATE_KIND_PRINT_DETAIL	Print Detail message template
13	TEMPLATE_KIND_EGALAXYSERVER_EMAIL	eGalaxy Server e-mail message template

## 14 Manager Workstation

## 14.1 GXUserKeywords

This table contains information for User Defined Keywords.

### Indexes and Constraints

Primary Key: PKUserKeywordID

Indexes:

(None)

Column	Type	Allow Nulls	Description
UserKeywordID	Int	N	Primary key, always unique
KeywordID	Int	N	Keyword id, as used by AX1180 and Galaxy POS
Description	char(60)	N	Keyword description
GroupID	Int	N	Group index field
Value	char(60)	N	Value for keyword

## 14.2 GxUsers

This table contains information such as ID, Password, Name, etc., for each person authorized to use Galaxy. It is equivalent to the "AGENTS.DAT" table in AX1180.

### Columns

Column	Type	Allow Nulls	Description
GxUserID	Int	N	Primary key, always unique
UserID	Int	N	The UserID as it is currently in AX1180 and Galaxy POS
Name	Char(20)	N	Name
Password	VarChar(128)	N	Password. This is stored in an encrypted form
NickName	Char(10)	N	This is a shortened version of your name
ClassID	Char(10)	N	This is a code value identifying what user class you belong to. This class controls your access privileges.
Expiration	Datetime	N	The expiration date of the current password. A date of 12/31/2079 means there is no expiration date.
Remaining	Int	N	The number of times you can login after your password has expired
UserName	Char(24)	N	This is your name, as it is used for logins
FailedRemaining	Int	N	This is the number of failed login attempts you were allowed before you are "locked-out" of the system. You are "locked-out" when the LockOut column is set to True.
LockOut	Bit	N	Flag to indicate that the user is not allowed to login
LastLogon	DateTime	Y	Date user last logged on to system
RequirePasswordChange	Bit	Y	Whether or not the user must change their password when they next log on
PersistenceState	Int	Y	Designates if this is local/Btree or sql and whether DBSync/WebPublishing will truncate the record <sup>1</sup>
Publish	Bit	Y	Determines if we should publish this user
ContactID	Int	Y	References CustContacts.CustContactID
ResetSecurityAnswer	Bit	Y	If the ResetSecurityAnswer is set to True (1), the user logging into the Reseller Web Store will need to provide an answer to their security question. This flag is ONLY used on the Reseller Web Store.
Active	Bit	Y	Indicates active users for Reseller Module - defaults to True.
SaltValue	Char(10)	Y	Salt value used for one-way password hashing
PasswordEncryptionMethod	Int	Y	Encryption method used for password where 0/null = legacy encryption and 1 = one-way hash.
LogonCardID	Varchar(100)	Y	The encrypted logon card ID for this user. The use of logon cards is an optional feature, so this column will often be blank/NULL.
LogonCardSaltValue	Varchar(10)	Y	The salt value used to encrypt the logon card ID number
GxUserGUID	UniqueIdentifier	Y	Alternate Primary Key, Always Unique
ExternalLogonName	Nvarchar(255)	Y	Identifier used by user to login to an external domain. Store users User Principal Name from LDAP
ExternalLogonID	Nvarchar(255)	Y	Unique Identifier for an external logon.

### Indexes

Name	Kind	Columns	Purpose
PKGxUsersGxUserID	P	GxUserID	Primary key
AXGxUsersUserID	A	UserID	To make sure the UserID is unique.
IXGxUsersContactID	IX	ContactID	For querying for contacts associated to a user
IXGxUsersClassID	IX	ClassID	For querying agency controls associated to a user
IXExternalLogonName	IX	ExternalLogonName	Speed up searching. Used during authentication
IXExternalLogonID	IX	ExternalLogonID	Speed up searching for external logon accounts.

<sup>1</sup> PersistenceState Values

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore GxUser records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store GxUsers.

### 14.3 Messages

Upon opening the system for the day, various messages can be setup that are selected by the users to read throughout the day. The **Messages** table contains these messages. Up to 25 messages can be defined.

#### Columns

Column	Type	Allow Nulls	Description
MessageID	Int	N	Primary key, always unique.
Descr	Char(26)	Y	Description of the message.
Lines	Char(200)	Y	Text contents of the message.

#### Indexes

Name	Kind	Columns	Purpose
PKMessagesMessageID	P	MessageID	Primary Key.

## 14.4 MWSAlerts

This table contains information on Access Control Point alerts.

### Columns

Column	Type	Allow Nulls	Description
AlertID	Int	N	Primary key; always unique. This is the "page_no" field in AX1180
AlertTime	DateTime	N	
AlertMsg	nVarChar(255)	N	
ACPID	Int	N	Access Control Point number
NodeID	Int	N	Node number
Location	nVarChar(24)	N	ACP name or Node name (if no ACP ID)
TimeOfDay	DateTime	N	
Acked	Bit	N	
Kind <sup>1</sup>	Int	N	
DisplayMsg	char(255)	N	
UserID	Int	N	
Sender	nVarChar(50)	Y	The sender of the message. Defined by the user sending the message. Populated only when Kind = ALERT_NODE_TO_NODE.
SenderMachineName	nVarChar(15)	Y	The NetBIOS name of the computer that sent the message. This is only used to determine which instance of MWS sent a message so that each MWS instance displays only messages sent by that MWS instances.
DialogType <sup>2</sup>	Integer	Y	The type of dialog box to use when displaying the message. Populated only when Kind = ALERT_NODE_TO_NODE.
DeliveryConfirmation	bit	Y	1 if a delivery confirmation should be sent by the Alert Notification service. When this is 1, the Alert Notification service uses the TCP/IP or NetBIOS network to notify the sender of the alert that the alert was acknowledged. The sender then receives a realtime update that the alert was displayed. Populated only when Kind = ALERT_NODE_TO_NODE.
DisplayDate	DateTime	Y	The date that the message should be displayed by the Alert Notification service. Populated only when Kind = ALERT_NODE_TO_NODE.
ExpirationDate	DateTime	Y	The date that the message expires. Populated only when Kind = ALERT_NODE_TO_NODE.
ExpireEndOfDay	bit	Y	1 if the message expires at end of day. The ExpirationDate column stores that expiration date. Populated only when Kind = ALERT_NODE_TO_NODE.
RecipientNodeID	Integer	Y	The node that will receive the message. Populated only when Kind = ALERT_NODE_TO_NODE.
RecipientUserID	Integer	Y	Once the message is displayed to a user, this is the user that the message was displayed to. Populated only when Kind = ALERT_NODE_TO_NODE.
Loaded	bit	Y	1 if the message was loaded into memory by the Alert Notification service. Populated only when Kind = ALERT_NODE_TO_NODE. If a message is not to be displayed until a date or time in the future, this will be set to 1 before the Displayed bit is set to 1.
Displayed	bit	Y	1 if the message was displayed to a user by the Alert Notification service. Populated only when Kind = ALERT_NODE_TO_NODE. Once a message has been displayed it will not be displayed again.
TimeDisplayed	DateTime	Y	The date and time that the message was displayed by the Alert Notification service. Populated only when Kind = ALERT_NODE_TO_NODE.

### Indexes

Name	Kind	Columns	Purpose
PKMWSAlertsAlertID	P	AlertID	Primary Key
IXSenderMachineName		SenderMachineName	Speed up queries
IXRecNodeIDExpDateDisplayed		RecipientNodeID, ExpirationDate, Displayed	Speed up queries
IXRecNodeIDExpDateLoaded		RecipientNodeID, ExpirationDate, Loaded	Speed up queries
IXACPAckedRecNodeKind		AcPId, Acked, Kind, RecipientNodeID	Speed up queries

### <sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	ALERT_NODE_OFFLINE	
2	ALERT_LEAD_PAGE	
3	ALERT_LOCKOUT	
4	ALERT_INACTIVE_TICKET	An alert generated for an inactive ticket validation
5	ALERT_RECEIPT_PRINTER	Printer is out of receipt paper
6	ALERT_TICKET_PRINTER	Printer is out of ticket stock
7	ALERT_KIOSK_OFFLINE	Kiosk is Offline
8	ALERT_KIOSK_CREDIT_CARD	Problem processing Credit Card, i.e. invalid credit card, expired credit card, etc.
9	ALERT_KIOSK_TRANSACTION	Error occurred during a Kiosk transaction.
10	ALERT_MSERVER_ERROR	Error occurred in the Mserver service.
11	ALERT_HSSERVER_ERROR	Error occurred in the HSserver service.
12	ALERT_LOCKERSERVER_ERROR	Error occurred in the LockerServer service.
13	ALERT_OLE_ERROR	Error occurred in the COM DLL.
14	ALERT_NODE_TO_NODE	This alert is a node-to-node message. Such messages are processed by the Alert Notification service. Currently only messages sent from MWS to other nodes are of this kind.

### <sup>2</sup> DialogType Values

Value	Gateway Constant Name	Description
0	dtInformation	Information
1	dtWarning	Warning
2	dtError	Error
3	dtConfirm	Confirm

## 14.5 MWSOperations

This table contains information on which nodes the Manager's Workstation Controller will send jobs to. This is similar to the AX1180 LeadOps table, except that it can contain entries for Access Control Point (turnstile) jobs, as well as POS.

### Indexes and Constraints

Primary Key: PKMWSOperationID

Indexes:

(None)

Column	Type	Allow Nulls	Description
MWSOperationID	Int	N	Primary key; always unique
NodeID	int	N	The Node number this record applies to.
ControllerNodeID	int	N	The Node Number of the Manager's Workstation Controller assigned to this Operation record. This is for future expansion. Currently, it is assumed that only one Controller will be in operation.
ResetShift	bit	N	
SendUsers	bit	N	Allows sending GXUsers information to this POS node
StationStatus	bit	N	
SendMessages	bit	N	Allows sending Messages information to this POS node
SendXRates	bit	N	Allows sending exchange rates to this POS node
SendWarnings	bit	N	Allows sending warnings information to this POS node
ChangeOrders	bit	N	Allows sending and receiving change order info to/from this POS node
CloseRequired	bit	N	
SendKeyword	bit	N	Allows sending user-defined keywords to this POS node.

## 14.6 PasswordHistory

This table contains the passwords used by all the users in the system.

### Indexes and Constraints

Primary Key: PasswordHistoryID

Indexes: UserID

The communication between the Galaxy and the Controller is time sensitive. If the Galaxy requests for a password change and the Controller times out, the Galaxy thinks that the Controller is off-line. It is essential that Controller retrieves a user's password history quickly to validate the new password.

Column	Type	Allow Nulls	Description
PasswordHistoryID	Int	N	Primary key, always unique.
GxUserID	Int	N	External key to GxUsers table.
UserID	Int	N	The UserID as it is currently in AX1180 and Galaxy POS.
Password	VarChar(128)	N	Password. This is stored in an encrypted form.
ChangedDate	DateTime	N	The date the Password was changed (new password).
ExpiredDate	DateTime	N	The system set expiration date of previous password.
SaltValue	Char(10)	Y	Salt value used for one-way password hashing
PasswordEncryptionMethod	Int	Y	Encryption method used for password where 0/null = legacy encryption and 1 = one-way hash.

## 15 Order Entry

### 15.1 ActiveCustCategories

This table is used to define the Active Customer Categories for any/all agencies. Display of customers or orders can be restricted to a set of customers by adding active customer categories for one or more agencies in this table. Only those customers or orders will be displayed that has an active customer category defined in this table for a given agency. All the other customers (customer with inactive customer categories) will not be displayed on any node of that agency.

#### Columns

Column	Type	Allow Nulls	Description
ActiveCustCategoryID	Int	N	Primary key, always unique.
AgencyID	Int	N	Agency number, Foreign key to Agencies.AgencyID
CategoryID	Int	N	Foreign key to CustCategories.CustCategoryID

#### Indexes

Name	Kind	Columns	Purpose
PKActiveCustCategoriesACCID	P	ActiveCustCategoryID	Primary Key.
IXActCustCatAgencyCategoryID		AgencyID, CategoryID	To improve query performance while finding the Active Customer Categories for a given agency.

## 15.2 Addresses

This table stores address information referenced by the **Customers** and the **CustContacts** tables.

### Columns

Column	Type	Allow Nulls	Description
AddressID	Int	N	Primary key, always unique.
AddressGUID	Uniqueidentifier	Y	GUID associated to this address. Used to uniquely identify this address across any system.
AddressType	Int	N	Code representing the type of address. <sup>1</sup>
OwnerId	Int	Y	No longer used.
Street1	VarChar(255)	N	Street address, first line.
Street2	VarChar(255)	Y	Street address, second line.
Street3	VarChar(255)	Y	Street address, third line.
City	VarChar(40)	N	City.
State	VarChar(40)	N	State / Province.
Postal	Char(16)	N	Postal code (ZIP code for USA).
CountryCode	Char(2)	Y	Country Code.
AddressCorrection	Bit	N	If TRUE, indicates that an external application modified this address record.
PersistenceState	Int	Y	Defines when this entry in the table should be deleted during a publish. On the Galaxy side, this column will always be null/zero. When addresses are saved to the web store SQL database during a publish, this value will be set to 3 (see values below). Before applying a publish, WebPublishing will "truncate" the Addresses table by deleting only entries with a PersistenceState of 3. <sup>2</sup>
AllowMailings	Bit	Y	Support ability to save Contact's preferences regarding communication methods. Indicates whether or not the Contact may be contacted via this address. Default value is True.

### Indexes

Name	Kind	Columns	Purpose
PKAddressesAddressID	P	AddressID	Primary key.
IXAddressesPostal	F	Postal	Used in the customer picklist.
IXAddressesCity		City	Used in the customer picklist.
IXAddressesState		State	Used in the customer picklist.
IXAddressesCountry		Country	Used in the customer picklist.

### <sup>1</sup> AddressType Values

Value	Description
1	Customer address.
2	Ship-to address. <i>This value is not currently used by the system.</i>
3	Bill-to address. <i>This value is not currently used by the system.</i>
4	Company address. <i>This value is not currently used by the system.</i>
5	Contact address.
6	Pass address

### <sup>2</sup> PersistenceState Values

Galaxy Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore Address records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store Addresses.

### 15.3 ARAccounts

This table holds billing information for customers. While many customers can use the same account to pay for purchases, only one customer is invoiced for the account's outstanding balance, and is said to "Own" the account. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
AccountID	Int	N	Primary key, always unique.
OwnerID	Int	N	Foreign key to Customers.CustomerID. This identifies the Customer who pays the bills for this account. Must contain a value greater than zero, and point to a valid customer.
BillToContactID	Int	N	Foreign key to CustContacts.CustContactID. This identifies the person to whom Invoices are sent. Must contain a value greater than zero, and point to a valid customer contact.
TermID	Int	N	Foreign key to ARTerms.TermID. This identifies the payment terms for this account.
StatementFormatID	Int	Y	This Column is currently not used by the system. Always has a value of zero.
CreditLimit	Money	N	Maximum currency amount the customer can charge to the account.
CreditHoldFlag	Bit	N	If TRUE, indicates that the customer's credit is on hold. If it is, the customer's credit privileges are temporarily revoked.
DaysToAutoHold	Int	Y	The number of days past due allowed for any invoice before the account is automatically placed on credit hold. A value of zero disables this feature. <i>This column is not currently used by the system.</i>
DaysDue	Int	N	Days to be added to the invoice date to determine the invoice due date.
CreditStatus	Int	Y	The status of the credit on the account <sup>1</sup>
PrepaidBalance	Money	Y	The balance paid in advance into the account. This money is available to be used for order payments.
DisableCredit	Bit	Y	Indicates if credit is disabled for the account. If this option is true, credit cannot be given to this account.
IsPrepaymentAccount	Bit	Y	This indicates if the account can be used as a prepayment account.
UsedCreditAmount	Money	Y	This field is comprised of the amount of issued and quoted tickets on unpaid orders which reduce a customer's available credit balance.
UsingUsedCreditAmount	Bit	Y	If this field is TRUE, a credit account balance is computed using a combination of the UsedCreditAmount, UsedOECreditAmount, and UsedPOSCreditAmount fields. If this field is set to FALSE, a credit account balance is computed using order balances, invoice balances and credit memo balances.
UsedOECreditAmount	Money	Y	This field is comprised of the amount of issued tickets on unpaid orders which reduce a customer's available credit balance.
UsedPOSCreditAmount	Money	Y	This field represents the amount used for unpaid Reseller Web Store transactions. If the "Enforce Credit Limit when adding items to an order" option is enabled in Order Configuration, the credit balance is calculated by combining this field with the UsedCreditAmount field. If the "Enforce Credit Limit when Issuing Ticket" option is enabled in Order Configuration, the credit balance is calculated by combining this field with the UsedOECreditAmount field.

#### Indexes

Name	Kind	Columns	Purpose
PKARAccountsAccountID	P	AccountID	Primary key.
IXARAccountsOwner	F	OwnerID	Needed to obtain the customer name from the Customers table.

#### <sup>1</sup> CreditStatus Values

Value	Gateway Constant Name	Description
0	ACCOUNT_STATUS_CURRENT	All transaction allowed.
1	ACCOUNT_STATUS_HOLD	No transactions allowed.
3	ACCOUNT_STATUS_FREEZE	Purchases allowed to draw down pre-paid balance; no other transactions allowed.

LastUpdatedBy values may include:

Value	Description
OrderUpdTr	Record was updated by the Update Trigger on the Orders table
OrderInsTr	Record was updated by the Insert Trigger on the Orders table
JnlDtInsTr	Record was updated by the Insert Trigger on the JnlDetails table

## 15.4 ARCreditMemoLines

This table stores the contents of a Credit Memo, including information on tickets, retail items, notes, payments (in this case, to the customer), and taxes.

### Columns

Column	Type	Allow Nulls	Description
CreditMemoLineID	Int	N	Primary key, always unique.
Description	Char(40)	Y	The item description, payment FOP description, or tax name, depending on the record's detail type.
OrderID	Int	Y	Foreign key to Orders.OrderID.
CreditMemoid	Int	Y	Foreign key to ARCreditMemos.CreditMemoid.
NoteID	Int	Y	Foreign key to Notes.NoteID.
DetailTableID	Int	Y	Identifies where more information on this record can be found, depending on the value of the DetailType <sup>1</sup> .
DetailType	Int	N	The type of detail record <sup>1</sup> .
Quantity	Int	Y	Quantity for this line.
IssuedQuantity	Int	Y	Number of issued items for this line.
PLU	Char(20)	Y	An alphanumeric code uniquely identifying a particular ticket or retail item. <sup>1</sup>
Amount	Money	Y	The unit price for this line. This is the price from the items table <b>with all taxes removed</b> (this is true even for tax-included tickets or items). The discount amount has <b>not</b> been removed from this price.  When the line's DetailType indicates that this is a "tax" line (DetailType = 6), this column will contain the total tax amount for previously processed credit memo lines.
Total	Money	Y	Total amount for all items in this line, using the calculation:  ((Amount + TaxAmount - Discount Amount) * Quantity).
DiscountAmount	Money	Y	Discount amount per ticket/item for this credit memo line. To determine what sort of discount is contained here, see the PriceBasis column.
EventID	Int	Y	Foreign key to RMEvents.EventID.
ReservationID	Int	Y	Foreign key to RMReservations.ReservationID.
RentalSerialID	Int	Y	Foreign key to RentalSerial.RentalSerialUniqueID.
HaveRentalInventory	Bit	N	If TRUE, indicates that this line has a Rental Inventory.
ResourceID	Int	Y	Foreign key to RMResources.ResourceID.
PrintNote	Bit	N	If TRUE, indicates that, if this credit memo line contains a note, the note can be printed on credit memo statements.
LineNbr	Int	Y	Unique number used to set the display sequence of credit memo lines within each credit memo. <b>Should only be altered by the system.</b>
TaxAmount	Money	Y	This is the amount of tax calculated on each unit of the ticket or item, whether it is tax-included or tax-excluded. However, any transactional taxes that may be applied to this ticket or item will not be included here. Transactional taxes will appear in separate "tax" credit memo lines.  When the line's DetailType indicates that this is a "tax" line (DetailType = 6), this column will contain the total transactional tax amount for previously processed credit memo lines.
PriceBasis	Int	Y	This value indicates what is contained in the DiscountAmount column <sup>2</sup>
TicketDate	DateTime	Y	Ticket date for date-specific tickets.
TransNode	int	Y	Node number from whose Journal transaction this line originated. Used to populate the @DTL_XXXX keywords for a credit memo.
TransNo	int	Y	Transaction number from the Journal transaction from which this line originated. Used to populate the @DTL_XXXX keywords for a credit memo.
TransDate	DateTime	Y	Transaction date from the Journal transaction from which this line originated. Used to populate the @DTL_XXXX keywords for a credit memo.
SalesProgramID	Integer	Y	Sales Program for the Credit Memo Line
GroupID	Integer	Y	UniqueID of the grouping credit memo line.
BaseID	Integer	Y	Used to group related CreditMemoLines/InvoiceLines together = CreditMemoLineID/InvoiceLineID of the primary CreditMemoLine/InvoiceLine for the group
BookingReference	NVarChar(40)	Y	Optional booking information recorded for reseller tickets
TravelDate	DateTime	Y	Optional travel date information recorded for reseller tickets

### Indexes

Name	Kind	Columns	Purpose
PKCMLinesCreditMemoLineID	P	CreditMemoLineID	Primary Key.
IXARCMLinesCreditMemoid	F	CreditMemoid	Used to select the lines of a particular credit memo.
IXARCMLinesOrderID	F	OrderID	Used to select credit memo lines associated with a particular order.
IXARCMLinesLineNbr		LineNbr	Used to sort the lines of a credit memo.

### <sup>1</sup> DetailType Values

Value	Description	Detail Table ID
1	Ticket/item detail record.	Foreign key to OrderDetails.OrderDetailID.
2	Payment detail record (payment applied to the credit memo).	Foreign key to ARCreditMemoPayments.CreditMemoPaymentID.
3	Note detail record.	Always zero
4	Return payment detail record (payment applied to a return on the original order).	Foreign key to OrderPayments.OrderPaymentID.
5	Order payment detail record (payment applied to the original order).	Foreign key to OrderPayments.OrderPaymentID.
6	Tax detail record.	The Tax ID, currently 1 thru 8.

### <sup>2</sup> Price Basis Values

Value	Description
0	Normal price. No discount has been applied, and the DiscountAmount should be zero.
1	A Sales Program has been applied to the line. The DiscountAmount contains the Sales Program rate amount.

2 The price has been manually changed. The DiscountAmount reflects the difference between the regular price and the price to which the line was changed.

## 15.5 ARCreditMemoPayments

This table stores information for payments made to a customer for a Credit Memo.

### Columns

Column	Type	Allow Nulls	Description
CreditMemoPaymentID	Int	N	Primary key, always unique.
CreditMemoLineID	Int	N	Foreign key to ARCreditMemoLines.CreditMemoLineID. This is the line in the credit memo to which this payment is attached.
PaymentDate	DateTime	Y	The date the payment was made to the invoice.
PaymentFOP	Int	Y	The form of payment used to make the payment.
PaymentAmount	Money	Y	The amount of the payment.
Endorsement	VarChar(50)	Y	The credit card or check endorsement.
ExpirationDate	Char(4)	Y	The credit card expiration date.
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.

### Indexes

Name	Kind	Columns	Purpose
PKARCMPaymentCMPaymentID	P	CreditMemoPaymentID	Primary Key.
IXARCreditMemoPaymentsLastUpdate		LastUpdate	Optional Index to improve performance of DB Update Process in DeleteCCInfoUtility.
IXARCreditMemoPaymentsInvoiceAgingReport		CreditMemoLineID, PaymentFOP, Endorsement, PaymentAmount, PaymentDate	Invoice Aging Report

## 15.6 ARCreditMemos

This table stores billing information for an order, supplying "header" information for Credit Memos. Credit Memo detail can be found in [ARCreditMemoLines](#)

### Columns

Column	Type	Allow Nulls	Description
CreditMemoid	Int	N	Primary key, always unique.
CustomerID	Int	N	Foreign key to Customers.CustomerID, representing the customer who created the Order(s) to which this Credit Memo applies.
OrderID	Int	Y	Foreign key to Orders.OrderID. When multiple orders are attached to a Credit Memo, this reflects the last Order attached.
MultipleOrders	Bit	N	If TRUE, indicates that more than one order is attached to this Credit Memo.
AccountID	Int	Y	Foreign key to ARAccounts.AccountID.
BillToContactID	Int	Y	Foreign key to CustContacts.CustContactID.
ShipToContactID	Int	Y	Foreign key to CustContacts.CustContactID.
PO	Char(20)	Y	The Purchase Order number of the original Order. When multiple orders are attached to a Credit Memo, this reflects the last Order attached.
CreditMemoStatus	Int	Y	The current Open status of the Credit Memo <sup>1</sup> .
OpenDate	DateTime	Y	The date the Credit Memo was created.
ClosedDate	DateTime	Y	The date the Credit Memo was closed. Defaults to 12/31/2079.
CreditMemoAmt	Money	Y	The original currency amount of the Credit Memo. This amount does not change after the credit memo is created.
Balance	Money	Y	The currently-due currency amount of the Credit Memo. This amount changes as payments are made.
DateDue	DateTime	Y	The date that the Credit Memo is required to be paid in full.
TermID	Int	Y	Foreign key to ARTerms.TermID, indicating the payment terms of the Credit Memo.

### Indexes

Name	Kind	Columns	Purpose
PKARCreditMemosCreditMemoid	P	CreditMemoid	Primary Key.
IXARCreditMemosCustomer	F	CustomerID	Used to select a list of credit memos by customer name or account.
IXARCreditMemosAccount	F	AccountID	Used to select a list of credit memos by account.
IXARCreditMemosStatus		CreditMemoStatus	Used to get a list of all open or closed credit memos.
IXARCreditMemosOpenDate		OpenDate	Used to select a list of credit memos that fall within a particular range of open dates.
IXARCreditMemosDueDate		DateDue	Used to select a list of credit memos that fall within a particular range of due dates.
IXARCreditMemosInvoiceAgingReport		OpenDate, CreditMemoAmt, DateDue, Balance, CustomerID, TermID	Invoice Aging Report

### <sup>1</sup> Status Values

Value	Description
1	The Credit Memo is Open
2	The Credit Memo is Closed

## 15.7 ARInvoiceLines

This table stores the contents of a Invoice, including information on tickets, retail items, notes, payments, and taxes. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
InvoiceLineID	Int	N	Primary key, always unique.
Description	Char(40)	Y	The item description, payment FOP description, or tax name, depending on the record's detail type.
OrderID	Int	Y	Foreign key to Orders.OrderID.
InvoiceID	Int	Y	Foreign key to ARInvoices.InvoiceID.
NoteID	Int	Y	Foreign key to Notes.NoteID.
DetailTableID	Int	Y	Identifies where more information on this record can be found, depending on the value of the DetailType <sup>1</sup> .
DetailType	Int	N	The type of detail record. <sup>1</sup>
Quantity	Int	Y	Quantity for this line.
IssuedQuantity	Int	Y	Number of issued tickets or retail items for this line.
PLU	Char(20)	Y	An alphanumeric code uniquely identifying a particular ticket or retail item. <sup>1</sup>
Amount	Money	Y	The unit price for this line. This is the price from the items table <b>with all taxes removed</b> (this is true even for tax-included tickets or items). The discount amount has <b>not</b> been removed from this price.  When the line's DetailType indicates that this is a "tax" line, this column will contain the total tax amount for previously processed invoice lines.
Total	Money	Y	Total amount for all items in this line, using the calculation:  ((Amount + TaxAmount - Discount Amount) * Quantity).
DiscountAmount	Money	Y	Discount amount per ticket/item for this invoice line. To determine what sort of discount is contained here, see the PriceBasis column.
EventID	Int	Y	Foreign key to RMEvent.EventID.
ReservationID	Int	Y	Foreign key to RMReservation.ReservationID.
RentalSerialID	Int	Y	Foreign key to theRentalSerial.RentalSerialID.
HaveRentalInventory	Bit	N	If TRUE, indicates that this line has a Rental Inventory.
ResourceID	Int	Y	Foreign key to RMResources.ResourceID.
TicketDate	DateTime	Y	Ticket date for date-specific tickets.
TransNode	int	Y	Node number from whose Journal transaction this line originated. Used when invoicing customer transactions from the Journal, without an order.
TransNo	int	Y	Transaction number from the Journal transaction from which this line originated. Used when invoicing customer transactions from the Journal, without an order.
TransDate	DateTime	Y	Transaction date from the Journal transaction from which this line originated. Used when invoicing customer transactions from the Journal, without an order.
PrintNote	Bit	N	If TRUE, indicates that, if this invoice line contains a note, the note can be printed on invoice statements.
CreditMemoid	Int	Y	Foreign key to ARCreditMemo.CreditMemo. This indicates that a Credit Memo has been applied to this invoice line.
LineNbr	Int	Y	Unique number used to set the display sequence of invoice lines within each invoice. <b>Should only be altered by the system.</b>
TaxAmount	Money	Y	This is the amount of tax calculated on each unit of the ticket or item, whether it is tax-included or tax-excluded. However, any transactional taxes that may be applied to this ticket or item will not be included in this column. Transactional taxes will appear in separate "tax" invoice lines, for orders only.  When the line's DetailType indicates that this is a "tax" line and the line has an OrderID, this column will contain the total transactional tax amount for previously processed invoice lines.
PriceBasis	Int	Y	This value indicates what is contained in the DiscountAmount column. <sup>2</sup>
SalesProgramID	Integer	Y	Sales Program for the Invoice Line
GroupID	Integer	Y	UniqueID of the grouping invoice line.
BaseID	Integer	Y	Used to group related CreditMemoLines/InvoiceLines together = CreditMemoLineID/InvoiceLineID of the primary CreditMemoLine/InvoiceLine for the group
BookingReference	NVarChar(40)	Y	Optional booking information recorded for reseller tickets
TravelDate	DateTime	Y	Optional travel date information recorded for reseller tickets

### Indexes

Name	Kind	Columns	Purpose
PKARInvoiceLinesInvoiceLinesID	P	InvoiceLineID	Primary Key.
IXARInvoiceLinesInvoiceID	F	InvoiceID	Used to select the lines of a particular invoice.
IXARInvoiceLinesOrderID	F	OrderID	Used to select invoice lines associated with a particular order.
IXARInvoiceLinesLineNbr		LineNbr	Used to sort the lines of an invoice.
IXARInvoiceLinesJnlTrans		TransNode, TransNo, TransDate	Used when creating an invoice for customer transactions that do not have an order associated with them.

<sup>1</sup> DetailType Values

Value	Description	Detail Table ID
1	Ticket/item detail record.	Foreign key to OrderDetails.OrderDetailID.
2	Payment detail record (payment applied to the invoice).	Foreign key to ARInvoicePayments.InvoicePaymentID.
3	Note detail record.	Always zero.
4	Return payment detail record (payment applied to a return on the original order).	Foreign key to OrderPayments.OrderPaymentID.
5	Order payment detail record (payment applied to the original order).	Foreign key to OrderPayments.OrderPaymentID.
6	Tax detail record.	The Tax ID, currently 1 thru 8.

<sup>2</sup> Price Basis Values

Value	Description
0	Normal price. No discount has been applied, and the DiscountAmount should be zero.
1	A Sales Program has been applied to the line. The DiscountAmount contains the Sales Program rate amount.
2	The price has been manually changed. The DiscountAmount reflects the difference between the regular price and the price to which the line was changed.

## 15.8 ArInvoiceOrders

This table stores a link from an invoice to an order.

Columns

Column	Type	Allow Nulls	Description
ARInvoiceOrderID	Int	N	Primary key, always unique
InvoiceID	Int	N	FK reference to ARInvoices.invoiceID column
OrderID	Int	N	FK reference to Orders.OrderID column

Indexes

Name	Kind	Columns	Purpose
PKARInvoiceOrderID	P	ARInvoiceOrderID	Primary Key.
IXARInvoiceOrdersInvIDOrdID		InvoiceID, OrderID	Used by the query to get all Orders linked to a given invoice

## 15.9 ARInvoicePayments

This table stores information on payments made by customers on an invoice. It is used by Order Entry, and Galaxy POS when configured to use SQL Customers.

### Columns

Column	Type	Allow Nulls	Description
InvoicePaymentID	Int	N	Primary key, always unique.
InvoiceLineID	Int	N	Foreign key to ARInvoiceLine.InvoiceLineID.
PaymentDate	DateTime	Y	The date the payment was made to the invoice.
PaymentFOP	Int	Y	The form of payment used to make the payment.
PaymentAmount	Money	Y	The amount of the payment.
CreditMemoID	Int	Y	Foreign key to ARCreditMemo.CreditMemoID. If the form of payment is a credit memo, this contains the credit memo number.
Endorsement	VarChar(50)	Y	The endorsement, if the payment is by credit card or check.
ExpirationDate	Char(4)	Y	The expiration date, if the payment is by credit card.
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.
SearchEndorsementValue	VarChar(50)	Y	This field stores last 4 characters of Endorsement. Will be encrypted if Encryption has been enabled and GxKeyID is not zero.

### Indexes

Name	Kind	Columns	Purpose
PKARInvoicePaymentsInvPayID	P	InvoicePaymentID	Primary Key.
IXARInvoicePaymentsLastUpdate		LastUpdate	Optional Index to improve performance of DB Update Process in DeleteCCInfoUtility.
IXARInvoicePymntsInvoiceLineID	F	InvoiceLineID	Optional index to improve performance of DB when executing a query that searches for invoices in Order Entry.
IXARInvoicePaymentsInvoiceAgingReport		InvoiceLineID, PaymentDate, PaymentFOP, PaymentAmount, Endorsement	Invoice Aging Report

## 15.10 ARInvoices

This table stores billing information for an order, supplying "header" information for Invoices. Invoice detail can be found in **ARInvoiceLines**. It is used by Order Entry, and Galaxy POS when configured to use SQL Customers.

### Columns

Column	Type	Allow Nulls	Description
InvoiceID	Int	N	Primary key, always unique.
CustomerID	Int	N	Foreign key to Customers.CustomerID, representing the customer who created the Order(s) this Invoice applies to.
OrderID	Int	Y	Foreign key to Orders.OrderID. When multiple orders are attached to an Invoice, this reflects the last Order attached.
MultipleOrders	Bit	N	If TRUE, indicates that more than one Order is attached to this Invoice.
AccountID	Int	Y	Foreign key to ARAccount.AccountID.
BillToContactID	Int	Y	Foreign key to CustContacts.ContactID.
ShipToContactID	Int	Y	Foreign key to CustContacts.ContactID.
PO	Char(20)	Y	The Purchase Order number of the original Order. When multiple orders are attached to an Invoice, this reflects the last order attached.
InvoiceStatus	Int	Y	The current Open status of the Invoice <sup>1</sup> .
OpenDate	DateTime	Y	The date the Invoice was created.
ClosedDate	DateTime	Y	The date the Invoice was closed.
InvoiceAmt	Money	Y	The original currency amount of the invoice. This amount does not change after the invoice is created.
Balance	Money	Y	The currently-due currency amount of the invoice. This amount changes as payments are made.
DateDue	DateTime	Y	The date that the invoice is required to be paid in full.
TermID	Int	Y	Foreign key to ARTerms.TermID, indicating the payment terms of the Invoice.

### Indexes

Name	Kind	Columns	Purpose
PKARInvoicesInvoiceID	P	InvoiceID	Primary Key.
IXARInvoicesCustomer	F	CustomerID	Used to select a list of invoices by customer name or account.
IXARInvoicesAccount	F	AccountID	Used to select a list of invoices by account.
IXARInvoicesStatus		InvoiceStatus	Used to get a list of all open or closed invoices.
IXARInvoicesOpenDate		OpenDate	Used to select a list of invoices that fall within a particular range of open dates.
IXARInvoicesDateDue		DateDue	Used to select a list of invoices that fall within a particular range of due dates.

<sup>1</sup> Status Values

Value	Description
1	The Invoice is Open
2	The Invoice is Closed

### 15.11 ARTerms

This table holds invoice payment information, such as due days and early payment discounts. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
TermID	Int	N	Primary key, always unique.
TermName	Char(10)	N	The name, or short description, of the Term.
Description	Char(50)	N	The long description of the term.
Type	Int	N	How the due date is calculated (currently 0, or "Net").
NetDays	Int	N	The number of days before payment is due (e.g. 10, 30, 60) for Net terms.
DiscountDays	Int	N	The numbers of days before the discount expires if there is a discount with the terms. <i>This column is currently not used by the system.</i>
DiscountRate	Float	N	The percentage rate of the discount if there is a discount with the terms. <i>This column is currently not used by the system.</i>
DepositAmount	Float	Y	Amount of required deposit for web group orders
DepositType	Integer	Y	Indicates whether DepositAmount is percentage or dollar amount

#### Indexes

Name	Kind	Columns	Purpose
PKARTermsTermID	P	TermID	Primary Key.
IXARTermsTermName		TermName	Used for sorting the Terms picklist by name.

## 15.12 CommunicationMethods

The CommunicationMethods table contains definition of a communication method. A communication method is assigned to a contact in Galaxy and used for the communication to the guest.

### Columns

Column	Type	Allow Nulls	Description
CommunicationMethodID	Int	N	Primary key, always unique
Name	Varchar(256)	N	Name of the communication method, e.g. E-mail, Phone, SMS
Description	Text	Y	Description of the communication method

### Indexes

Name	Kind	Columns	Purpose
PKCommunicationMethodID	P	CommunicationMethodID	Primary Key.

### 15.13 ContactConnections

This table provides a link between a contact and a customer. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
ContactConnectionID	Int	N	Primary key, always unique.
ContactID	Int	N	Foreign key to CustContacts.CustContactID.
ConnectionType	Int	N	Connection type value <sup>1</sup> .
ConnectionID	Int	N	Foreign key into the connection's table based on ConnectionType.
Preferred	Bit	Y	If set to 1, the communication method associated with the connection is the preferred method of communication for the contact
WebGroupSales	Bit	Y	When set to 1, indicates that the contact is for the GroupSales on the Web module
AllowChargeFOPs	Bit	Y	This field will indicate whether charge fops can be used by the contact when purchasing on the web - default value is False.
SystemLogonGenerated	Int	Y	<p>This field will indicate whether a given Contact Connection has had a System Logon generated.</p> <p>Values:</p> <p>0 = Not generated (default)</p> <p>1 = Processed Successfully</p> <p>2 = Processed With Error</p>

#### Indexes

Name	Kind	Columns	Purpose
PKContactConnectionsConConnID	P	ContactConnectionID	Primary Key.
IXContactConnectionsContactID	F	ContactID	Used to speed-up the Auto Batch Print process.
IXContactConnTypeConnID		ConnectionID, ConnectionType	Used to speed-up the Auto Batch Print process.
CXContactConnectionIDContactID		ConnectionID, ContactID	Used to speed-up the loading of primary contact for an order

<sup>1</sup> ConnectionType Values

Value	Gateway Constant Name	Description
1	ID_CUSTOMER_CONNECTION	Customer connection. ConnectionID is a foreign key to the Customers table
2	ID_COMMUNICATION_METHOD_CONNECTION	Communication method connection. ConnectionID is a foreign key to the CommunicationMethods table
3	ID_CARRIER_CONNECTION	Carriers connection. ConnectionID is a foreign key to the Carriers table
4	ID_GROUPSales_CUSTOMER_CONNECTION	GroupSales Customer Connection. ConnectionID is a foreign key to the Customers table.

### 15.14 ContactDonorTypes

ContactDonorTypes contains the connection between the contact and the code table lookup value ID.

#### Columns

Column	Type	Allow Nulls	Description
ContactDonorTypeID	Integer	N	Primary Key, always unique
ContactID	Integer	N	Foreign Key, references the CustContacts table
CodeTableValueID	Integer	N	Link to the code table value ID
ContactDonorTypeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKContactDonorTypeID	P	ContactDonorTypeID	Primary Key.

## 15.15 ContactLists

ContactLists contain header records for each list. A ContactList is a list of contacts. There are many use for a ContactList, one of them is to send promotional information to all contacts in the list. ContactList also allows to group similar contacts.

### Columns

Column	Type	Allow Nulls	Description
ContactListID	Integer	N	Primary Key, always unique
Name	Varchar(50)	N	Short name for the list
Description	Varchar(255)	Y	Long description of the list
ContactListGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKContactListContactListID	P	ContactListID	Primary key.

## 15.16 ContactListDetails

ContactListDetails contains the connection between the list header and the actual customer, contact or pass holder on the list.

### Columns

Column	Type	Allow Nulls	Description
ContactListDetailID	Integer	N	Primary Key, always unique
ContactListID	Integer	N	Foreign Key, references the ContactLists table
DetailType	Integer	N	Internal value indicating what is being linked to the list.
DetailID	Integer	N	Foreign Key, references the link to what is being added to the list
ContactListDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKContactListDtlsCntListDtlID	P	ContactListDetailID	Primary key.
IXContactListDtlsDtlTypeAndID		DetailType, DetailID	Used by the query to retrieve all contact lists for the given DetailType and DetailID
IXContactListDtlsContactListID		ContactListID	Used by the query to retrieve all contact list details for the given ContactListID

## 15.17 ContactPaymentInfo

The ContactPaymentInfo table is used to store the payment information for an individual.

### Columns

Column	Type	Allow Nulls	Description
ContactPaymentInfoID	Integer	N	Primary Key, always unique
ContactID	Integer	N	Foreign Key, references the unique identifier CustContacts.CustContactID. This refers back to the billing information of the contact.
CardNumber	VarChar(50)	N	The individual's card number
CardHolderName	VarChar(100)	Y	The card holder's full name
ExpirationDate	VarChar(20)	Y	The card's expiration date
IsPrimary	Bit	Y	Used to find the primary card for a particular Contact ID
PaymentName	VarChar(100)	Y	User defined name for this payment (i.e. "Personal Credit Card", "Corporate Card", etc.)
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.
PaymentType	Int	N	Specifies the type of payment method the record is using. <sup>1</sup>
BankAccountNumber	VarChar(40)	Y	Bank Account Number (IBAN) if the PaymentType=1. See PaymentType values below.
BankRoutingNumber	VarChar(40)	Y	Bank Routing Number (BIC) if the PaymentType=1. See PaymentType values below.
CardToken	NVarChar(62)	Y	CardToken
DataSource	Int	Y	The source from which the payment data was obtained. This indicates if the data was entered manually or obtained through a payment plugin. <sup>2</sup>
BankEncrypted	Bit	Y	Indicates if the BankAccountNumber and BankRoutingNumber fields are encrypted.

### Indexes

Name	Kind	Columns	Purpose
PKContactPaymentInfoID	P	ContactPaymentInfoID	Primary Key.
IXCPIContactIDLastUpdate	IX	ContactID, LastUpdate	Used by the Contract Processor export process to search for changes

### <sup>1</sup> PaymentType Values

Value	Gateway Constant Name	Description
0	PAYMENTTYPE_CREDITCARD	Payment Method is Credit Card, so the Credit Card-related fields are used
1	PAYMENTTYPE_BANKACCOUNT	Payment Method is Bank Account, so the Bank Account-related fields are used

### <sup>2</sup> DataSource Values

Value	Gateway Constant Name	Description
0	DATA_SOURCE_MANUAL	The payment data was entered by the user in Galaxy.
1	DATA_SOURCE_PLUGIN	The payment data was sent to Galaxy from a payment plugin.

## 15.18 CustCategories

Customers can be grouped together into logical collections, or categories. For example, all High School customers may be grouped together in a School category and High School subcategory. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
CustCategoryID	Int	N	Primary key, always unique.
ParentID	Int	N	Foreign key to CustCategories.CustCategoryID.
Description	Char(40)	N	Description for this customer category.
StatementID	Int	N	Foreign key to Statements.StatementID. This specifies the statement template to be used for order statements.
InvoiceStatementID	Int	N	Foreign key to Statements.StatementID. This specifies the statement template to be used for invoices.
RefundStatementID	Int	N	Foreign key to Statements.StatementID. This specifies the statement template to be used for credit memos.
BatchStatementID	Int	N	Foreign key to Statements.StatementID. This specifies the statement template to be used for batch printing.
CombineInvoices	Bit	N	If TRUE, indicates that one invoice should contain multiple Orders.
TaxSetID	Int	Y	Foreign key to TaxSets.TaxSetID. Provides a default tax set when assigning a customer category to a customer.
FopSetID	Int	Y	Foreign key to FopSets.FopSetID. Provides a default form of payment set when assigning a customer category to a customer.
OrderRuleID	Int	Y	Order Rule Identifier, Foreign key to OrderRules.OrderRuleID
AllowMultipleSalesPrograms	Bit	Y	Whether or not category allows multiple sales programs to be applied
EnforceLimits	Bit	Y	This flag will control whether or not to prompt the user with the date of visit and group size prompt at the start of an order.
MaxGroupQty	Int	Y	Stores the maximum number of groups allowed
MaxGuestQty	Int	Y	Stores the maximum number of guests allowed
AttributeGroupID	Int	Y	Points to the Attribute Values Group
CustomerFieldAttributeGroupID	Int	Y	FK reference to FieldAttributeGroups.FieldAttributeGroupID. Defines attributes to enforce when adding a customer in Galaxy.
TicketActivationStatementID	Int	Y	Foreign key to Statements.StatementID. This specifies the statement template to be used for ticket activation.
EnforceRatio	Bit	Y	This field is used to request that the order line quantity on tickets added to the order because of the associated ticket ratio is managed by Associated Ticket Engine only (read only to the user).
CommunicationGroupID	Uniqueidentifier	Y	A unique identifier that is used to associate records in the OrderCommunications table with a customer, customer category, or sales channel category group.
CustCategoryGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKCustCategoryCustCategoryID	P	CustCategoryID	Primary Key.
IXCustCategoryParentID	F	ParentID	Used to select the sub-categories of a category.
IXCustCategoriesOrderRuleID		OrderRuleID	To improve query performance while finding the Customer Categories for a given Order Rule.

### 15.19 CustCategoryTotals

Stores a running total of maximum of groups and maximum number of guests per group visit date.

#### Columns

Column	Type	Allow Nulls	Description
CustCategoryTotalID	Int	N	Primary key, always unique
CustCategoryID	Int	N	Customer Category ID
TotalGroupQty	Int	N	Sum total of all CustCategories for a given group visit date
TotalGuestQty	Int	N	Sum total of expected guests visiting the venue for a given group visit date.
GroupVisitDate	DateTime	N	Group visit date for an order/customer.

#### Indexes

Name	Kind	Columns	Purpose
PKCustCtgryTtlCustCtgryTtlID	P	CustCategoryTotalID	Primary Key

## 15.20 CustContacts

This table contains the information necessary to communicate with a particular customer, via postal mail, electronic mail, telephone, or fax machine. These contacts are specific to individual customers, and cannot be shared. It is used by Order Entry, and Galaxy POS when configured to use SQL Customers.

### Columns

Column	Type	Allow Nulls	Description
CustContactID	Int	N	Primary key, always unique.
CustContactGUID	Uniqueidentifier	Y	GUID associated to this contact. Used to uniquely identify this contact across any system.
ContactType	Int	N	Type of contact. <sup>1</sup>
JobTitle	Char(30)	Y	Contact's title or position (i.e. President, Senior Manager, etc.)
Salutation	Char(30)	Y	Greeting used in correspondence with this contact (i.e. Mr., Mrs., Ms., etc.) <b>NOTE</b>  If Galaxy loads a contact that contains a value in the Salutation column, it will convert it on-the-fly by performing the following steps:  1. Looks for an existing entry in the NameTitles table for the value in the Salutation field. 2. If one is found, CustContacts.NameTitleID is set to this NameTitle entry. Skip to step 4. 3. If a matching NameTitle is not found, create one and set CustContacts.NameTitleID is set to this NameTitle entry. 4. Clear the CustContacts.Salutation column. 5. Save the Contact.
FirstName	Char(30)	Y	Contact's first name.
MiddleName	Char(30)	Y	Contact's middle name.
LastName	Char(30)	N	Contact's last name.
UnformattedLastName	Char(30)	Y	A column stripped off of control chars fro LastName, like letter accents.
AddressID	Int	N	Foreign key to Addresses.AddressID.
Phone	Char(30)	Y	Contact's phone number.
UnformattedPhone	Varchar(30)	Y	A Copy of Phone without any formatting
Fax	Char(30)	Y	Contact's fax number.
Cell	Char(30)	Y	Contact's cell phone number.
Email	VarChar(128)	Y	Contact's email address.
PrimaryContact	Bit	N	If TRUE, indicates that this is the primary contact for this Contact Type.
NoteID	Int	Y	Foreign key to Notes.NoteID.
ExternalID	VarChar(64)	Y	Additional contact information relating to an external system.
NameTitleID	Int	Y	Foreign key to NameTitles.NameTitleID
NameSuffixID	Int	Y	Foreign key to NameSuffixes.NameSuffixID
DOB	Date/Time	Y	Date of Birth for this contact
Gender	Int	Y	Gender of this contact <sup>2</sup> Defaults to 0 = Unspecified
AgeGroup	Int	Y	The age group the contact falls within <sup>3</sup>
PersistenceState	Int	Y	For Web Publishing, see values <sup>4</sup>
IdentificationNo	Varchar(64)	Y	A personal identification number for the contact
AllowEmail	Bit	Y	Support ability to save Contact's preferences regarding communication methods. Indicates whether or not the Contact may be contacted via email. Default value is True.
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.
SpecialNeeds	Bit	Y	Indicates the Special Needs status of the contact.
Deceased	Bit	Y	Indicates if the contact is deceased.
DataErasure	int	Y	Status of the contacts request to have their data anonymized <sup>5</sup> .
LastFourIDNo	VarChar(4)	Y	The un-encrypted last four characters of the IdentificationNo column, for easy searching.

### Indexes

Name	Kind	Columns	Purpose
PKCustContactsCustContactID	P	CustContactID	Primary Key.
IXCustContactsAddress	F	AddressID	Used by the contacts picklist to search for contacts by city, state, postal code, or country
IXCustContactsLastName		LastName	Used by the contacts picklist to search for contacts by last name.
IXCustContactsMiddleName		MiddleName	Used by the contacts picklist to search for contacts by middle name.
IXCustContactsFirstName		FirstName	Used by the contacts picklist to search for contacts by first name.
IXCustContactsContactType		ContactType	Used by the contacts picklist to search for contacts by contact type.
IXCustContactsPhone		Phone	Used by the contacts picklist to search for contacts by phone number.
IXCustContactsExternalID		ExternalID	Index used by eGalaxy processing
IXCustContactsNameSuffixID		NameSuffixID	Search contact via NameSuffixID
IXCustContactsNameTitleID		NameTitleID	Search contact via NameTitleID
IXCustContactsUnformattedPhone		UnformattedPhone	Search by UnformattedPhone in Kiosk
IXCustContactsUnformattedLastName		UnformattedLastName	Search by UnformattedLastName in Kiosk
CXCustContactsCustContactIDPrimaryContact		CustContactID, PrimaryContact	Search by primary contact

IXCustContactsLastUpdate	CustContactID, LastUpdate	Used by the Contract Processor export process to search for changes for this table
IXCustContactsAddressNameDOB	AddressID, MiddleName, LastName, FirstName, DOB	To eliminate Pass Purchaser and Primary Query timeouts
IXCustContactsIdentificationNo	IdentificationNo	Used to find Contact record by IdentificationNo
IXCustContactsEmail	Email	Improve performance of query to open an email and view its contents from Web Store Maintenance.
IXLastFourIDNo	LastFourIDNo	Improve performance when querying by the LastFourIDNo column.

**1 ContactType Values**

Value	Gateway Constant Name	Description
1	ID_CUSTOMER_CONTACT	Customer Contact.
2	ID_ORDER_CONTACT	Order Contact. This value is not currently used by the system.
3	ID_BILLTO_CONTACT	Bill-to Contact.
4	ID_SHIPTO_CONTACT	Ship-to Contact.
5	ID_EMPLOYEE_INDIVIDUAL	Currently not used.
6	ID_PATRON_INDIVIDUAL	Currently not used.
7	ID_PASS_CONTACT	Pass Contact
8	ID_JOINT_MEMBER	Joint member
9	ID_ATTENDEE	Roster attendee
10	ID_CONTRACT_OWNER	Payment contract owner
11	ID_LOYALTY	Loyalty Contact
12	ID_EXPRESS_CONTACT	Express Busbill Contact
13	ID_GUEST_CONTACT	Guest Name Contact

**2 Gender Values**

Value	Gateway Constant Name	Description
0	GENDER_UNSPECIFIED	Contact's gender has not been specified
1	GENDER_MALE	Contact is a male
2	GENDER_FEMALE	Contact is a female

**3 AgeGroup Values**

Value	Gateway Constant Name	Description
0	JOINT_MEMBER_ADULT_AGE_GROUP	Contact is considered an adult
1	JOINT_MEMBER_CHILD_AGE_GROUP	Contact is considered a child

**4 PersistenceState Values**

Gateway Constant Name	Value	Description
PERSISTENCE_STATE_TRUNCATE	0	Designates sql records created by Galaxy (or another application) that can be truncated by DBSync processes.
PERSISTENCE_STATE_DO_NOT_TRUNCATE	1	Designates sql records created by MWS (or another application) that should NOT be truncated by DBSync processes. This is currently used to preserve Reseller UserProfiles and related records (profileprivs, profilecontrols, profile item groups and profile agency controls) that are only available in MWS.
PERSISTENCE_STATE_WEBSTORE_DO_NOT_TRUNCATE	2	Designates records created and maintained by the web store. Records are imported to the Galaxy side but should not be truncated on the web store or exported to the web store during web publishing process.
PERSISTENCE_STATE_PUBLISHED_FROM_GALAXY	3	This persistence state will be assigned to webstore custcontact records as they are published from Galaxy; these are the records that can be deleted and re-imported to the webstore during future publishing processes. No other records should be deleted from web store CustContacts.

**5 DataErasure Values**

Value	Description
NULL	Default Value for records created prior to EU_GDPR Script being run - No request has been made or attempted.
0	Default Value for records created after the EU_GDPR Script has been run - No request has been made or attempted.
1	Contact has requested for their data to be anonymized .
2	Contact data has been anonymized.
-	Below are error values stored when anonymizing the contact data has failed.
-2	Contact is associated with a record in PaymentContracts where CurrentBalance <> 0.
-3	Contact is the Bill To Contact associated with a record in ARAccounts where UsedOECreditAmount <> 0, UsedCreditAmount <> 0, or UsedPOSCreditAmount <> 0.
-4	Contact is the Bill To Contact associated with a record in ARInvoices where Balance <> 0.
-5	Contact is associated with a record in Orders where Balance <> 0.
-6	Contact is associated with a valid pass that is not expired.
-7	Contact is the Bill To Contact associated with a record in ARCreditMemos where Balance <> 0.

### 15.21 CustContactsConsent

This table contains an audit trail of the contacts consent to a newsletter.

#### Columns

Column	Type	Allow Nulls	Description
CustContactID	Varchar(64)	N	Foreign key to CustContacts.CustContactID.
OrderID	int	N	Foreign key to Orders.OrderID. References the order the audit record is linked to.
ExternalID	Nvarchar(64)	Y	External Order ID
FirstName	Nvarchar(30)	Y	Contact's first name.
MiddleName	Nvarchar(30)	Y	Contact's middle name.
LastName	Nvarchar(30)	Y	Contact's last name.
NewsletterOptIn	Nvarchar(max)	N	The contacts consent to receiving a newsletter.
ConsentDate	DateTime	N	Date the consent was recorded.
ConsentChannel	VARCHAR(10)	N	How the consent record was updated.
RemoteAddress	VARCHAR(20)	Y	Remote address of the order associated with the consent record.
MerchantID	int	Y	MerchantID the order was place through.
OptInText	NVarChar(max)	Y	The text that the contact gave consent to.

## 15.22 CustContactsConsentProcess

This table is a staging table for the consent to web orders and orders in order entry. This data is processed into the CustContactsConsent table off hours through a SQL Agent job.

### Columns

Column	Type	Allow Nulls	Description
CustContactConsentProcessID	bigint	N	Primary key, always unique.
OrderID	int	N	Foreign key to Orders.OrderID. References the order the audit record is linked to.
OEUserFieldID	int	N	Foreign key to OEUserFields.OEUserFieldID.
FieldSequence	int	N	Foreign key to GxUserFields.FieldSequence.
Response	Nvarchar(max)	N	Either contains the consent or the consent text based on the FieldSequence.
ConsentDate	DateTime	N	Date the consent was recorded.
Processed	int	N	Status of this record being processed into the CustContactsConsent table <sup>1</sup> .
ProcessedDateTime	DateTime	Y	Timestamp this record was processed.
LastUpdatedBy	Nvarchar(20)	Y	Node and Username triggering this change.

### Indexes

Name	Kind	Columns	Purpose
PKCustContactsConsentProcess	P	CustContactConsentProcessID	Primary Key.

<sup>1</sup> Processed Values

Value	Description
0	Unprocessed.
1	Currently Being Processed.
2	Processed.

## 15.23 Customers

The **Customers** table contains groups for which there is or will be frequent activity. Customer information is used in regular ticketing to track customer activity and record customer purchases made with credit. Customers are also used in Order Entry, for maintaining ticket orders over time (see **Orders** table).

### Columns

Column	Type	Allow Nulls	Description
CustomerID	Int	N	Primary key, always unique.
CategoryID	Int	N	Foreign key to CustCategories.CategoryID. Must contain a value greater than zero, and point to a valid customer category.
AccountID	Int	Y	Foreign key to ARAccounts.AccountID. This field is zero (0) or NULL if the customer does not have charge account privileges.
ExternalAccount	Char(20)	Y	Alphanumeric code external to the Order Entry system.
CustName	Char(50)	N	Customer's name.
AddressID	Int	N	Foreign key to Addresses.AddressID. Must contain a value greater than zero, and point to a valid address.
Phone	Char(30)	Y	Customer's Telephone Number.
UnformattedPhone	Varchar(30)	Y	A copy of the phone number without any formatting
Fax	Char(30)	Y	Customer's Fax Number.
CellPhone	Char(30)	Y	Customer's Cell phone number.
Email	Varchar(128)	Y	Customer's Email Address.
WebUrl	Char(128)	Y	Customer's World Wide Web address.
TaxExemptID	Char(30)	Y	Tax-exempt ID for non-taxable customers.
TaxSetID	Int	Y	Foreign key, reference to the TaxSets table.
LanguageID	Int	Y	Customer's native language. This column is not currently used by the system.
CombineInvoices	Bit	Y	If TRUE, indicates that one invoice should contain multiple Orders.
Reference	Char(40)	Y	Additional customer information.
UDFData1	Varchar(255)	Y	This field holds user-defined information.
UDFData2	Varchar(255)	Y	This field holds user-defined information.
UDFData3	Varchar(255)	Y	This field holds user-defined information.
UDFData4	Varchar(255)	Y	This field holds user-defined information.
UDFData5	Varchar(255)	Y	This field holds user-defined information.
UDFData6	Varchar(255)	Y	This field holds user-defined information.
UDFData7	Varchar(255)	Y	This field holds user-defined information.
UDFData8	Varchar(255)	Y	This field holds user-defined information.
NoteID	Int	Y	Foreign key to Notes.NoteID. This column is not currently used by the system.
Salesperson	Char(16)	Y	Salesman assigned to this customer.
FopSetID	Int	Y	Foreign key to FopSets.FopSetID.
LastActivity	DateTime	Y	The date and time of the last sale to this customer.
DefaultSalesProgram	Int	Y	The ID of the default Sales Program to use when the customer is selected in POS or Order Entry, or zero if none.
DefaultMenu	Char(16)	Y	The name of the default menu to use when the customer is selected in the main POS screen. The default menu is used if this is blank. Not used in Order Entry.
OrderPickupFromDays	Int	Y	Defines the number of days when the Quick Pickup Galaxy function can pick up orders for this customer before the Group Visitdate
OrderPickupThruDays	Int	Y	Defines the number of days when the Quick Pickup Galaxy function can pick up orders for this customer after the Group Visitdate
AllowMultipleSalesPrograms	Bit	Y	Whether or not category allows multiple sales programs to be applied
IgnoreContactConnections	Bit	Y	Flag to prevent contacts to be associated to the customer.
CompanyCode	Varchar(128)	Y	Code used by Group Sales module to associate an existing Customer to a web contact. Company code is distributed by the Galaxy system admin to the third-party customers so they can purchase special price tickets on the Webstore.
Status	Int	Y	The current status of the customer. <sup>1</sup>
PictureID	Int	Y	Link to Pictures.PictureID - supports the ability to link a jpg image file to the Customer, e.g., customer logo.
DisplayOrderInfo	Bit	Y	When this value is set to 1, ACS32 looks for order payment information when any ticket or pass sold to this customer is scanned at an ACP. If this value is set to 1, the GTS_SP_GetCreditCardPaymentInfo stored procedure must be defined in the database or TCON32 or Direct to Database ACS32 will record an error indicating that the stored procedure could not be executed.
CommunicationGroupID	UniqueIdentifier	Y	A unique identifier that is used to associate records in the OrderCommunications table with a customer, customer category, or sales channel category group.
CustomerGUID	UniqueIdentifier	Y	GUID associated to this customer. Used to uniquely identify this customer across any system.

### Indexes

Name	Kind	Columns	Purpose
PKCustomersCustomerID	P	CustomerID	Primary Key.
IXCustomersAccount	F	AccountID	Used by the customer picklist to display the owner of the account being used by a customer.
IXCustomersCategory	F	CategoryID	Used by the customer picklist to search for customers by category.
IXCustomersAddress	F	AddressID	Used by the customer picklist to search for customers by city, state, postal code, or country.
IXCustomersCustName		CustName	Used by the customer picklist to search for customers by name.
IXCustomersExternalAccount		ExternalAccount	Used by the customer picklist to search for customers by the external account.
IXCustomersSalesperson		Salesperson	Used by the customer picklist to search for customers by salesperson.
IXCustomersReference		Reference	Used by the customer picklist to search for customers by reference.
IXCustomersCustIDCatID		CustomerID, CategoryID	Optimizing customer search queries
IXCustomersCompanyCode		CompanyCode	To improve performance of query to get a customer associated with a GroupSales contact.
IXCustomerGUID		CustomerGUID	To improve performance of searching by CustomerGUID.

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	csValid	Customer is valid.
1	csPending	Customer is pending for approval.
2	csRejected	Customer is rejected during approval process.

## 15.24 DeliveryMethods

Contains definition of the delivery methods used in Order Entry. Options such as Shipping charge PLU, Estimated number of days, and instructions on how to fulfill a delivery method can also be configured on a delivery method.

### Columns

Column	Type	Allow Nulls	Description
DeliveryMethodID	Integer	N	Primary key. Obtained from GatewayCounters.
Kind	Integer	N	Kind of the delivery method or numeric identifier for the delivery method <sup>1</sup>
Name	Varchar(256)	N	Name of the delivery method
Description	Text	Y	Detailed description explaining what this delivery method does
LongDescription	Text	Y	Long description of the shipping method
ShippingChargePLU	Char(20)	Y	PLU used to apply shipping charges
Instructions	Text	Y	Instructions to display on the web store for a delivery method
EstimatedDays	Integer	Y	Estimated number of days for delivery of this shipping method. Used by the web store to calculate the value of keyword @EstimatedDeliveryDate
WebTemplateID	Integer	Y	FK reference to WebTemplates table  The message is displayed on confirmation page and email when this delivery method is selected by the guest on web store. Note that this template is sent to the <b>shipping</b> contact's email address.
Instructions	Text	Y	Instructions to display on the web store and confirmation e-mail for a delivery method
MarketingMessageID	Integer	Y	FK reference to WebTemplates table  Defines a marketing message to display when this delivery method is selected.
Identifier	Varchar(30)	Y	Used to hold a company identifier to help distinguish between similar delivery methods.
AllowShipMonday	Bit	Y	Include Monday when calculating shipping date using EstimatedDays
AllowShipTuesday	Bit	Y	Include Tuesday when calculating shipping date using EstimatedDays
AllowShipWednesday	Bit	Y	Include Wednesday when calculating shipping date using EstimatedDays
AllowShipThursday	Bit	Y	Include Thursday when calculating shipping date using EstimatedDays
AllowShipFriday	Bit	Y	Include Friday when calculating shipping date using EstimatedDays
AllowShipSaturday	Bit	Y	Include Saturday when calculating shipping date using EstimatedDays
AllowShipSunday	Bit	Y	Include Sunday when calculating shipping date using EstimatedDays
AllowBatchPrint	Bit	Y	Orders that use this delivery method can be batch printed
AllowKioskPickup	Bit	Y	Orders that use this delivery method can be picked up at a kiosk
AllowQuickOrderPickup	Bit	Y	Orders that use this delivery method can be picked up via Quick Order Pickup
AllowQOPCompTickets	Bit	Y	Orders that use this delivery method can issue comp tickets
BackupDeliveryMethodID	Int	Y	DeliveryMethodID of the Delivery Method that is to be used by the web store if displaying the pop-up print at home screen is unsuccessful. This is only used by the PrintOnWeb Delivery Method kind.
FieldAttributeGroupID	Int	Y	Foreign key to FieldAttributeGroups table
SeparatePrintAtHomeAttachments	Bit	Y	Used if Kind=6 (PrintAtHome), instructs WebOrderProcessor to generate separate PDF files for each ticket
BillingWebTemplateID	Int	Y	FK reference to WebTemplates table. The message is displayed on the confirmation page and an email sent to the billing contact's email address when this delivery method is selected by the guest on web store. WebTemplateID (existing column) description should probably be updated to note that it is the template sent to the <b>shipping</b> contact's email address.
MobileProvider	Int	Y	Valid only for "MMS/SMS" delivery method. Designates the MMS/SMS mobile service provider to be used when delivering messages using this delivery method. <sup>2</sup>
MobileMessageContent	Text	Y	Text of the SMS message. Gateway script is supported for Ticket and Order.
MobilePrintAtHome	TinyInt	N	Determines how to handle email confirmation when sending ticket data via SMS. Default value is 2. <sup>3</sup>
ImmediateIssue	Bit	Y	
AllowKioskReprint	Bit	Y	Orders that use this delivery method can have their tickets, passes and packages reprinted at the kiosk
CommunicationGroupID	uniqueidentifier	Y	Unique identifier used to associate records in the OrderCommunications table with this delivery method.
IssuanceType	Int	Y	Specifies when issuance occurs for an order with this delivery method - immediate (issuance occurs when order is received by eGalaxy Server, similar to Print on Web), delayed (issuance occurs in Web Order Processor, similar to Print at Home), or other (issuance occurs in OE, QOP, Kiosk, etc.). <sup>4</sup>
IssueAsInactive	Bit	Y	When using this delivery method, issue products with the status of 'Inactive' (3). Only available for 'Custom' delivery method kinds.
DeliveryMethodGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKDeliveryMethodsDelMethodID	P	DeliveryMethodID	Primary Key.
IXDeliveryMethodsKind		Kind	Index used by the query to get Shipping methods by kind

<sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
0	DM_NONE	No delivery method
1	DM_HOLD	Hold or WillCall
2	DM_MAIL	Mail
3	DM_OVERNIGHT	Overnight
4	DM_SECONDDAY	Second-Day
5	DM_OTHER	Other
6	DM_PRINTATHOME	Print at home

7	DM_KIOSK	Kiosk (pickup at the kiosk)
8	DM_MMS_SMS	MMS/SMS
9	DM_PRINT_ON_WEB	Print on Web
10	DM_A_LA_CARTE	A la Carte

**<sup>2</sup> MobileProvider Values**

Value	Gateway Constant Name	Description
0	MP_NONE	No mobile provider
1	MP_HENGRUI	Hengrui
2	MP_GREENTICKET	Gateway Ticketing Systems

**<sup>3</sup> MobilePrintAtHome Values**

Value	Gateway Constant Name	Description
0		No email is sent
1		Confirmation email is sent, but no Print@Home ticket
2		Confirmation email is sent with Print@Home ticket(s) attached

**<sup>4</sup> IssuanceType Values**

Value	Gateway Constant Name	Description
0		Immediate Issuance
1		Delayed Issuance
2		Other Issuance

## 15.25 DeliveryMethodGroups

DeliveryMethodGroups contains header records for each list. A DeliveryMethodGroup is a list of delivery methods. A DeliveryMethodGroup can be assigned to an item or ticket in order to enforce the methods of delivery.

### Columns

Column	Type	Allow Nulls	Description
DeliveryMethodGroupID	Integer	N	Primary key. Obtained from GatewayCounters.
Name	Varchar(30)	N	Name of the delivery method group
DeliveryMethodGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKDelIMGroupsDelIMGroupID	P	DeliveryMethodGroupID	Primary Key.

## 15.26 DeliveryMethodGroupDetails

DeliveryMethodGroupDetails contains the connection between the group header and the actual delivery method.

### Columns

Column	Type	Allow Nulls	Description
DeliveryMethodGroupDetailID	Integer	N	Primary key. Obtained from GatewayCounters.
DeliveryMethodGroupID	Integer	N	FK reference to DeliveryMethodGroups table
DeliveryMethodID	Integer	N	FK reference to DeliveryMethods table
DeliveryMethodGroupDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKDMGrpDtlsDMGrpDtlsID	P	DeliveryMethodGroupDetailID	Primary Key.
IXDMGrpDetailDelMethodGroupID		DeliveryMethodGroupID	Used by the query to get all delivery methods (DeliveryMethodID) for a given DeliveryMethodGroupID
IXDMGrpDetailDeliveryMethodID		DeliveryMethodID	Used by the query to get all details for a given DeliveryMethodID

## 15.27 FilterDefinitions

This table is used to hold the filters created in data maintenance.

### Columns

Column	Type	Allow Nulls	Description
FilterDefinitionID	Integer	N	Primary key
Name	Varchar(255)	N	Short name for the filter
FilterType	Integer	N	An internal value, designating the type of filter. <sup>1</sup>
FilterText	Text	Y	XML formatted filter criteria

### Indexes

Name	Kind	Columns	Purpose
PKFilterDefinitionFilterDefID	P	FilterDefinitionID	Primary
IXFilterDefinitionsFilterType		FilterType	Allow looks up by filter type faster

<sup>1</sup> FilterType Values

Value	Gateway Constant Name	Description
1	ID_ORDER_DEFINITION	Filters for Orders

## 15.28 FilterConnections

This table is used to hold the associations between filter definitions and any other SQL table. An example of this would be Agencies.

### Columns

Column	Type	Allow Nulls	Description
FilterConnectionID	Integer	N	Type of header record, see table.
FilterDefinitionID	Integer	N	Foreign key for FilterDefinitions
ConnectionID	Integer	N	The unique id of the record that is being associated to the filter.
ConnectionType	Integer	N	An interval value, designating the table that is being used to link with the filter definition. <sup>1</sup>
FunctionType	Integer	N	An internal value, designating where in the system the filter definition may be used. <sup>2</sup>

### Indexes

Name	Kind	Columns	Purpose
PKFilterConnectionFilterConID	P	FilterConnectionID	Primary key.
IXFilterConnectionsFilterDefID		FilterDefinitionID	Link to Filter Definitions

<sup>1</sup> ConnectionType Values

Value	Gateway Constant Name	Description
1	ID_AGENCY_CONNECTION	Value for connections to Agency ID

<sup>2</sup> FunctionType Values

Value	Gateway Constant Name	Description
1	ID_BATCH_PRINT_FUNCTION	Value for connection for the use in Batch Print

## 15.29 FOPSets

This table stores describes collections (or sets) of Forms of Payment. The Forms of Payment that are included in each of these "sets" can be found in the **FopSetDetails** table. This table is used by Order Entry, and by Galaxy POS when using SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
FopSetID	Int	N	Primary key, always unique.
Description	Char(20)	Y	Form of Payment Set description.
FopSetGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKFopSetsFopSetID	P	FopSetID	Primary Key.

### 15.30 FOPSetDetails

This table defines a Form of Payment for a Form of Payment Set. This table is used by Order Entry, and by Galaxy POS when using SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
FopSetDetailID	Int	N	Primary key, Always unique.
FopSetID	Int	Y	Foreign key to FopSets.FopSetID.
FopID	Int	Y	A Form of Payment ID Number, 10 thru 89.
Sequence	Int	N	Display sequence, used when displaying FOPs on the web store
FopSetDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKFopSetDetailsFopSetDetailID	P	FopSetDetailID	Primary Key.
IXFopSetDetailsFopSet	F	FopSetID	Used to select the detail records for a particular FOP set.

### 15.31 GroupSalesCodes

The GroupSalesCodes table contains the codes used in group sales to associate a Contact or an order with a Customer.

#### Columns required now

Column	Type	Allow Nulls	Description
GroupSalesCodeID	Int	N	Unique number to identify the GroupSales Code and customer combination.
Code	VarChar(30)	N	Alphanumeric code, defined on the Customer in Galaxy Maintenance, but stored here. It is used by a Contact placing an order on the web to provide Customer-specific pricing.
CustomerID	Int	N	Foreign key into Customers.CustomerID. The customer for which this code is used. Note that one customer may have several codes.

#### Indexes

Name	Kind	Columns	Purpose
PKGroupSalesCodeID	P	<i>GroupSalesCodeID</i>	Primary Key.
AKGroupSalesCodesCode	A	<i>Code</i>	Unique key to enforce uniqueness of Code.
IXGroupSalesCodesCustomerID		<i>CustomerID</i>	Index to facilitate query to retrieve all codes associated with a CustomerID

## 15.32 GXUserFields

This table defines generic user-defined fields, including the prompt (or label), type, size, and purpose of the information required. Currently used by Order Entry for user-defined Order fields.

### Columns

Column	Type	Allow Nulls	Description
GXUserFieldID	Int	N	Primary key, always unique.
Kind	Int	Y	Denotes whether the user defined field is used for Orders, Customers, Passes, etc.. <sup>1</sup> Currently, this is always 1 (Orders).
RefID	Int	N	Foreign key to PassKinds.ID, when the Kind is set to Passes. <i>This column is not currently used by the system.</i>
FieldLabel	Char(30)	Y	Text label for the screen prompt.
FieldKind	Int	Y	Denotes the data type of response, i.e. numeric, text, etc.. <sup>2</sup>
FieldLength	Int	Y	Length of response field.
DecimalPlaces	Int	Y	Number of decimals if response type is floating point or currency.
DataRequired	Bit	Y	If TRUE, indicates that a response to the field is required before saving the user-defined field record.
ClearForNew	Bit	Y	If TRUE, indicates that the field is to be cleared when making recurring entries.
SwipeAction	Int	Y	Denotes response when card swipe occurs in this field. <sup>3</sup>
DefaultValue	Char(30)	Y	Default value for field.
MinValue	Char(30)	Y	Minimum Value for range checking on the field.
MaxValue	Char(30)	Y	Maximum Value for range checking on the field.
MchoicelD	Int	Y	ID value from Multiple Choice table, null if multiple choice not used.
HelpText	Char(80)	Y	Text to be displayed as a hint on the screen when user places mouse cursor over the field.
FieldSequence	Int	N	The order in which the fields will appear on the screen.
PrivRequired	Bit	Y	If TRUE, indicates that a user is required to have special privileges to make changes to the field. <i>This column is not currently used by the system.</i>

### Indexes

Name	Kind	Columns	Purpose
PKGXUserFieldsGXUserFieldID	P	GXUserFieldID	Primary Key.
IXGXUserFieldsFieldSequence		FieldSequence	Needed to speed-up the Auto Batch Print process.

### <sup>1</sup> Kind Values

Value	Description
1	Orders: Responses are stored in the OEUserFields table.

### <sup>2</sup> FieldKind Values

Value	Description
0	Numeric
1	Text
2	Logical (True or False)
3	Multiple Choice

### <sup>3</sup> SwipeAction Values

Value	Description
0	None
1	Write
2	Overwrite

### 15.33 ItemConstraints

This table stores the tickets and retail items that can be sold as part of a **SalesProgram**. It is currently used by Order Entry, and by Galaxy POS when configured to use SQL **Customers**. An item constraint can apply to either a single PLU number (a ticket or retail item), or to up to four "groups" of PLU's.

#### Columns

Column	Type	Allow Nulls	Description
ItemConstraintID	Int	N	Primary key, always unique.
SalesProgramID	Int	Y	Foreign key to SalesPrograms.SalesProgramID. This identifies the sales program, of which this constraint is a part
PLU	Char(20)	N	An alphanumeric code uniquely identifying a particular ticket or retail item. <sup>1</sup>
ItemGroup1	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the constraint applies. <sup>2</sup>
ItemGroup2	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the constraint applies. <sup>2</sup>
ItemGroup3	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the constraint applies. <sup>2</sup>
ItemGroup4	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the constraint applies. <sup>2</sup>
ItemConstraintGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemConstraintsItmConstrntID	P	ItemConstraintID	Primary Key.
IXItemConstraintsSPItemGroup1		SalesProgramID, ItemGroup1	Reduce query time
IXItemConstraintsSPItemGroup2		SalesProgramID, ItemGroup2	Reduce query time
IXItemConstraintsSPItemGroup3		SalesProgramID, ItemGroup3	Reduce query time
IXItemConstraintsSPItemGroup4		SalesProgramID, ItemGroup4	Reduce query time

<sup>1</sup> The PLU column should be blank if the constraints apply to a group of PLUs (rather than a single PLU).

<sup>2</sup> This column should have a value of zero (0) or NULL if the constraint applies to a single PLU.

### 15.34 ItemDeliveryConversions

This table stores data to allow automatic conversions from one PLU to another based on delivery method when processing an order through eGalaxy.

#### Columns

Column	Type	Allow Nulls	Description
ItemDeliveryConversionID	Int	N	Primary key, always unique
BasePLU	Char(20)	N	PLU that requires replacement by delivery method
DeliveryMethodID	Int	N	The unique ID of the delivery method that will cause a replacement of the base PLU with the replacement PLU
ReplacementPLU	Char(20)	N	PLU that replaces the base PLU when the delivery method used on the order matches the DeliveryMethodID

#### Indexes

Name	Kind	Columns	Purpose
PKItemDeliveryConversionID	P	ItemDeliveryConversionID	Primary Key.
IXItemDeliveryConversionsBasePLU	IX	BasePLU	Enhance speed of selecting list of conversions for a given PLU

### 15.35 ItemGroupDetails

This table stores the contents of **ItemGroups**. An Item Group is a collection of tickets and/or items. This table is currently used by **ItemConstraints** and **ItemRates**, as part of a **SalesProgram**.

#### Columns

Column	Type	Allow Nulls	Description
ItemGroupDetailID	Int	N	Primary key, always unique.
ItemGroupID	Int	N	Foreign key to ItemGroups.ItemGroupID, representing the Item Group of which the record is a member.
PLU	Char(20)	N	An alphanumeric code uniquely identifying a particular ticket or retail item.
ItemGroupDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemGroupDetailsItemGrpDtlID	P	ItemGroupDetailID	Primary Key.
IIXItemGroupDetailsItemGroupID		ItemGroupID	Used to select the details of a specific item group.

### 15.36 ItemGroups

An Item Group is a collection of tickets and/or items. This table is currently used by **ItemConstraints** and **ItemRates**, as part of a **SalesProgram**. The contents of an Item Group can be found in the **ItemGroupDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
ItemGroupID	Int	N	Primary key, always unique.
Description	Char(40)	N	The description of this Item Group.
Inactive	Bit	N	True if Item Group is Inactive, and not visible on most picklists.
ItemGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKItemGroupsItemGroupID	P	ItemGroupID	Primary Key.

### 15.37 ItemRates

An Item Rate contains the pricing information for a **SalesProgram**. This table is currently used by Order Entry, and by Galaxy when configured for SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
ItemRateID	Int	N	Primary key, always unique.
SalesProgramID	Int	N	Foreign key to SalesPrograms.SalesProgramID, representing the Sales Program of which this rate is a member.
PLU	Char(20)	N	An alphanumeric code uniquely identifying a particular ticket or retail item. <sup>1</sup>
ItemGroup1	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the rate applies. <sup>2</sup>
ItemGroup2	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the rate applies. <sup>2</sup>
ItemGroup3	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the rate applies. <sup>2</sup>
ItemGroup4	Int	Y	Foreign key to ItemGroups.ItemGroupID. This identifies a collection of PLU numbers (tickets and/or retail items) to which the rate applies. <sup>2</sup>
MinQuantity	Int	Y	The minimum purchase quantity to qualify for the rate to apply.
MaxQuantity	Int	Y	The maximum quantity of tickets or items to discount/change. <sup>3</sup>
RateType	Int	N	Indicates how the price is calculated. <sup>5</sup>
Amount	Float	N	The actual percent or currency amount to apply to the price of the PLU, depending on the RateType. <sup>5</sup>
Priority	Float	Y	When multiple rates are used for the same PLU in a Sales Program, this determines the order in which each Rate is applied. <sup>4</sup>
LastRate	Bit	N	If TRUE, indicates that no other rate (with a lower or equal priority) will apply. Currently, always set to TRUE.
IsTableBased	Bit	N	If TRUE, indicates that this rate is a table-based one.
MinQtyOverride	Bit	Y	Use the total quantity of selected items from the sales program's item constraints as the minimum quantities when determining when rates should be applied.
ItemRateGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems
PriceProgramGroupMethod	Int	Y	Indicates where to obtain the Price Program Group that is used to calculate the price when the RateType is 4 (ITEM_RATE_TYPE_PRICE_SCHEDULE). <sup>6</sup>
PriceProgramGroupID	Int	Y	The Price Program Group to use for calculating the price when the RateType is 4 (ITEM_RATE_TYPE_PRICE_SCHEDULE) and PriceProgramGroupMethod is 1 (ITEM_RATE_PRICE_PROGRAM_GROUP_METHOD_RATE).

#### Indexes

Name	Kind	Columns	Purpose
PKItemRatesItemRateID	P	ItemRateID	Primary Key.
IXItemRatesSalesProgramID		SalesProgramID	Used to select the item rates for a particular sales program.

<sup>1</sup> The PLU column should be blank if the rate applies to a group of PLUs (rather than a single PLU).

<sup>2</sup> This column should have a value of zero (0) or NULL if the rate applies to a single PLU.

<sup>3</sup> If the MaxQuantity field is zero (0), the rate will apply to an unlimited number of tickets or items.

<sup>4</sup> The Priority must be between 0.01 (lowest priority) and 100 (highest priority).

<sup>5</sup> RateType Values

Value	Gateway Constant	Description	Amount
1	ITEM_RATE_TYPE_PERCENT	Percentage discount	The percentage (12.34 = 12.34%) discount to apply to the price of the ticket or retail item.
2	ITEM_RATE_TYPE_RATE	Currency amount discount	The amount to subtract from the price of the ticket or retail item.
3	ITEM_RATE_TYPE_VALUE	Flat price	The amount to charge the customer. Replaces the price of the ticket or retail item.
4	ITEM_RATE_TYPE_PRICE_SCHEDULE	Price determined by price schedule	The amount to charge the customer is determined from the Price Schedule configured on the sales program. Replaces the price of the ticket or retail item.

<sup>6</sup> PriceProgramGroupMethod Values

Value	Gateway Constant	Description
0/NULL	ITEM_RATE_PRICE_PROGRAM_GROUP_METHOD_PRODUCT	Legacy behavior. Inherit the Price Program Group from the Product. ItemRates.PriceProgramGroupID is ignored.
1	ITEM_RATE_PRICE_PROGRAM_GROUP_METHOD_RATE	Use the Price Program Group defined on the Item Rate, ItemRates.PriceProgramGroupID.

**15.38 OEBatchPrintDetails**

This table lists the tickets on an order that will be printed as part of a batch ("Batch Printing").

**Columns**

Column	Type	Allow Nulls	Description
BatchDetailsId	Int	N	Primary key, always unique.
BatchHeaderId	Int	Y	Foreign key to OEBatchPrintHeader.BatchHeaderID, defining the batch of which this record is a member.
OrderId	Int	Y	Foreign key to Orders.OrderID, specifying the order from which to print tickets.
OrderLineId	Int	N	Foreign key to OrderLines.OrderLineID, specifying the order line from which to print tickets.
PrintPriority	Int	Y	The sequence in which the tickets are printed.
PrintPartialOrder	Bit	Y	If TRUE, indicates that only the tickets listed here are to be printed, rather than all tickets for the order.
PLU	Char(20)	Y	An alphanumeric code uniquely identifying a particular ticket or retail item.
TicketDate	DateTime	Y	This is the date if the ticket is a Date Specific Ticket.
QtyToPrint	Int	Y	The number of tickets to print.
BaseID	Int	Y	Used to link Item Fee detail to the Item that required the fee
DetailType	Int	Y	Same DetailType on the order line for which this batch print detail was created - for values see OrderLines Table

**Indexes**

Name	Kind	Columns	Purpose
PKOEBatchPrintDtlsBatDetailID	P	BatchDetailsId	Primary Key.
IXOEBatchPrintDtlsOrderID	F	OrderId	Used to select the batch print detail records for a particular order.
IXOEBatchPrintDtlsDtIIDRecVer		BatchDetailsID, RecordVersion	Used to speed-up the Auto Batch Print process.
IXOEBatchPrintDtlsHdrPriority		BatchHeaderId, PrintPriority	Used to select the batch print detail records for a particular batch header. It is also used to sort batch print details by batch header and priority.

### 15.39 OEBatchPrintHeader

Order tickets can be printed as part of a batch ("Batch Printing"). This table defines the batches which contain those tickets. A batch can contain all the tickets on an order, or only some of them. The tickets themselves are listed in the **OEBatchPrintDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
BatchHeaderID	Int	N	Primary key, always unique.
Description	Char(40)	Y	Batch Header description.
Node	Int	Y	POS Node that will print this batch.
Status	Int	Y	Current status of the Batch. <sup>1</sup>
AutoPrint	Bit	Y	If TRUE, indicates that the batch will be automatically printed without user intervention.

#### Indexes

Name	Kind	Columns	Purpose
PKOEBatchPrintHeaderBatchHdrID	P	BatchHeaderID	Primary Key.
IXOEBatchPrintHeaderNode		Node	Used to select the batches for a particular node for Auto Batch Print.

<sup>1</sup> Status Values

Value	Description
1	Normal
2	Printing

### 15.40 OECreditAccountUpdateLog

This table is used in the GTS\_OE\_UpdateJnlUsedCreditAmountOnAccount stored procedure to log updates to the ARAccounts table.

#### Columns

Column	Type	Allow Nulls	Description
OECreditAccountUpdateLogID	Int	N	Primary key, always unique (Identity)
JnlTranID	Int	N	FK to JnlHeaders, the journal transaction associated with this update
JnlPaymentID	Int	N	Foreign key to JnlPayments.PaymentID
AccountID	Int	N	Foreign key to ARAccounts.AccountID
FOPID	Int	N	Foreign key to FOPS.FOPCode
TransactionKind	Int	N	Type of transaction (0 = OE, 1 = POS)
PaymentAmount	Money	N	Amount paid
CurrentUsedOECreditAmount	Money	N	Current value of UsedOECreditAmount on the account, prior to this update
NewUsedOECreditAmount	Money	N	New value of UsedOECreditAmount on the account, after this update
CurrentUsedPOSCreditAmount	Money	N	Current value of UsedPOSCreditAmount on the account, prior to this update
NewUsedPOSCreditAmount	Money	N	New value of UsedPOSCreditAmount on the account, after this update
LogDateTime	DateTime	N	Date/time of the log entry

#### Indexes

Name	Kind	Columns	Purpose
PKOECreditAccountUpdateLogID	P	OECreditAccountUpdateLogID	Primary Key

**15.41 OEGroupVisits**

This table stores the guest arrival information for orders.

**Columns**

Column	Type	Allow Nulls	Description
GroupVisitID	Int	N	Primary key, always unique.
CustomerID	Int	Y	Foreign key to Customers.CustomerID, representing the person or persons to be visiting (this may be different from the customer generating the order, or paying the bill). This may be zero or NULL if the Description field is used instead.
VisitDate	DateTime	N	The date and time the group is expected to arrive.
DepartureDate	DateTime	Y	The date and time the group is expected to leave the facility
Reference	VarChar(255)	Y	A note or reference field.
Description	Char(50)	Y	The name or description of the person or persons visiting. This may be used as an alternative to selecting an existing customer.
ExpectedNumGuests	Int	Y	The total number of people expected to arrive.
NoteID	Int	Y	Foreign key to Notes.NoteID. <i>This column is not currently used by the system.</i>

**Indexes**

Name	Kind	Columns	Purpose
PKOEGroupVisitsGroupVisitID	P	GroupVisitID	Primary Key.
IXOEGroupVisitsCustomerID	F	CustomerID	Used to select customer information for a particular group.
IXOEGroupVisitsArrivalSummary		Visit date	Arrival Summary Report

## 15.42 OEGuests

This table stores information of the arriving Guest for Auto Batch Print.

### Columns

Column	Type	Allow Nulls	Description
OEGuestID	Int	N	Primary key, always unique.
ExternalID	Int	Y	Foreign key, referencing tables to be determined. <i>This column is not currently used by the system.</i>
Title	Char(30)	Y	Title or position of the Guest (i.e. President, Senior Manager, etc.)
Salutation	Char(15)	Y	Greeting used in correspondence with the Guest (i.e. Mr., Mrs., Ms., etc.)
FirstName	Char(30)	N	First name of the Guest.
MiddleName	VarChar(30)	Y	Middle Initial of the Guest.
LastName	Char(30)	Y	Last Name of the Guest.
Phone	Char(25)	Y	Phone number of the Guest.
Fax	Char(25)	Y	Fax number if the Guest.
AddressID	Int	Y	Foreign key to Addresses.AddressID.
Email	VarChar(128)	Y	Email address of the Guest.
OrderLineID	Int	Y	Foreign key to Orderlines.OrderLineID - the orderline this guest is associated with.
ContactID	Int	Y	Foreign key to CustContacts.CustContactID

### Indexes

Name	Kind	Columns	Purpose
PKOEGuestsGuestID	P	OEGuestID	Primary Key.
IXOEGuestsFirstName		FirstName	Used to improve query performance while inserting orderliness.
IXOEGuestsLastNam		LastName	Used to improve performance of Order Manager when searching by Last Name
IXOEGuestsContactID		ContactID	Foreign key to CustContacts.CustContactID
IXOEGuestsOrderLineID		OrderLineID	Foreign key to OrderLines.OrderLineID

### 15.43 OEMarkedOrders

This table stores Orders that have been marked for Invoice or Credit Memo generation.

#### Columns

Column	Type	Allow Nulls	Description
MarkedOrderID	Int	N	Primary key, always unique.
OrderID	Int	N	Foreign key to Orders.OrderID, representing the order to be processed.
MarkedStatus	Int	N	Indicates why the order was marked. <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKOEMarkedOrdersMarkedOrderID	P	MarkedOrderID	Primary Key.
IXOEMarkedOrdersOrderIDStatus		OrderID, MarkedStatus	Used to select a list of orders for a particular order status.

<sup>1</sup> **MarkedStatus Values**

Value	Gateway Constant Name	Description
1	ID_INVOICE_MARKEDORDER	The order was marked for invoicing.
2	ID_CREDITMEMO_MARKEDORDER	The order was marked for a credit memo.
3	ID_BATCHPRINT_MARKEDORDER	The order was marked for Batch Print.
4	ID_SUSPENDED_MARKEDORDER	The order was marked for Suspended Batch Print
5	ID_CLOSE_MARKEDORDER	The order was marked for closure.
6	ID_DELETE_MARKEDORDER	The order was marked for delete.
7	ID_PRINT_MARKEDORDER	The order was marked for print.
8	ID_RECALC_MARKEDORDER	The order was marked to be recalculated.
9	ID_ACTIVATE_MARKEDORDER	The order was marked to have its tickets activated.
10	ID_DEACTIVATE_MARKEDORDER	The order was marked to have its tickets deactivated.
11	ID_LOCK_MARKEDORDER	The order was marked to be locked.
12	ID_UNLOCK_MARKEDORDER	The order was marked to be unlocked.

## 15.44 OStatementGroups

Contains statements defined on a customer category or on order configuration. One or more statements can be defined for the customer category as well as for the order configuration.

### Columns

Column	Type	Allow Nulls	Description
OEStatementGroupId	Integer	N	Primary key. Obtained from GatewayCounters
GroupIDType	Integer	N	Identifies the contents of the GroupID, see table. <sup>1</sup>
GroupID	Integer	N	A numeric code uniquely identifying the entity to whom this statement belong to
StatementID	Integer	N	FK reference to Statements table
StatementType	Integer	N	Type of the statement, see table. <sup>2</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOStatementGrpsOEStatGrpID	P	OEStatementGroupId	Primary key.
IXOStatGrpTypeGrpIDStatType		GroupIDType, GroupID, StatementType	Index, used by query to retrieve statements for the given GroupType, GroupID, and StatementType

### <sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	Description
0	ID_CUST_CATEGORY	Statement is defined on the customer category. The GroupID is the CustCategoryID
1	ID_ORDER_CONFIG	Statement is defined on the order configurations. The GroupID is the Node number (of the Order Entry node)

### <sup>2</sup> StatementType Values

Value	Gateway Constant Name	Description
0	GENERAL_STATEMENT	General statement. Can be printed for any order from the Order Entry screen

### 15.45 OESuspendedOrders

This table contains information on orders that were interrupted during Batch Printing. This does *not* include orders whose printing was aborted.

#### Columns

Column	Type	Allow Nulls	Description
OrderID	Int	N	Primary key value from the Orders table, always unique <sup>1</sup>
TransID	Int	Y <sup>2</sup>	The number of the transaction that was suspended
SuspendedDate	DateTime	Y <sup>2</sup>	The date the batch print was suspended
QtyPrinted	Int	Y <sup>2</sup>	The number of tickets that were printed before the batch print was suspended
Agent	Int	Y <sup>2</sup>	The agent number of the person suspending the batch print
Node	Int	Y <sup>2</sup>	The node number from which the batch print was suspended

<sup>1</sup> The OrderID is a reference to the order in the Orders table that was suspended.

<sup>2</sup> Although these fields are not necessary for the suspending and resuming of batch printed orders to function properly, they must be non-null in order for the data to be displayed on the "Suspended Orders Report".

#### Indexes

None

## 15.46 OEUserFields

This table is used by Order Entry to store the response to User Defines Fields for orders. The field itself is defined in [GXUserFields](#).

### Columns

Column	Type	Allow Nulls	Description
OEUserFieldID	Int	N	Primary key, always unique.
GxUserFieldID	Int	N	Foreign key to GXUserFields.GXUserFieldID.
OrderID	Int	N	Foreign key to Orders.OrderID.
Response	NVarChar(max)	Y	Data in response to the User-Defined Field.

### Indexes

Name	Kind	Columns	Purpose
PKOEUserFieldsOEUserFieldID	P	OEUserFieldID	Primary Key.
IXOEUserFieldsOrderGxUserFld		OrderID, GxUserFieldID	Used to speed-up the Auto Batch Print process.

## 15.47 OEWebTemplateGroups

The OEWebTemplateGroups table contains groups of web templates defined at either the Customer Category or Order Configuration level. Entries in this table define what web templates are available from the drop-down list on the Email button on the toolbar of the main Order Entry screen.

### Columns

Column	Type	Allow Nulls	Description
OEWebTemplateGroupID	Integer	N	Primary key. Obtained from GatewayCounters
GroupIDType	Integer	N	Identifies the contents of the GroupID <sup>1</sup>
GroupID	Integer	N	A numeric code uniquely identifying the entity to whom this web template belongs to
WebTemplateID	Integer	N	FK reference to Statements table
WebTemplateType	Integer	N	Type of the web template <sup>2</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOEWebTGrpOEWebTemplateGrpID	P	OEWebTemplateGroupID	Primary key
IXOEWBETGrpTypeGrpIDWebTType		GroupIDType, GroupID, WebTemplateType	Index, used by query to retrieve statements for the given GroupType, GroupID, and WebTemplateType

<sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	Description
0	CUST_CATEGORY_GROUPID_TYPE	Web Template is defined on the customer category. The GroupID is the CustCategoryID
1	ORDER_CONFIG_GROUPID_TYPE	Web Template is defined in the order configuration. The GroupID is the Node number (of the Order Entry node)

<sup>2</sup> WebTemplateType Values

Value	Gateway Constant Name	Description
0	GENERAL_WEB_TEMPLATE_TYPE	General web template. Can be generated for any order from the Order Entry screen

## 15.48 OrderCommands

This table stores the commands to be processed by Auto Batch Print.

### Columns

Column	Type	Allow Nulls	Description
OrderCommandID	Int	N	Primary key, always unique.
OrderID	Int	Y	Foreign key to Orders.OrderID.
Action <sup>1</sup>	Int	N	The type <sup>1</sup> of the Order Command to be performed.
Node	Int	N	Node, where the command will be processed.
BatchNo	Int	Y	Foreign key to OEBatchPrintHeader.BatchHeaderID, referencing the batch to be printed.
Status <sup>2</sup>	Int	Y	Status of the OrderCommand
ExternalID	Varchar(128)	Y	ExternalID for the order command. At present used to store ExternalID of the eGalaxySource

### Indexes

Name	Kind	Columns	Purpose
PKOrderCommandOrderCommandID	P	OrderCommandID	Primary Key.
IXOrderCommandIDRecVer		OrderCommandID, RecordVersion	Used to speed-up the Auto Batch Print process.
IXOrderCommandNodeBatchAction		Node, BatchNo, Action	Used to speed-up the Auto Batch Print process.

### <sup>1</sup> Action Values

Value	Gateway Constant Name	Description
1	DELETE_ORDER_COMMAND_ACTION	Delete
2	PRINT_ORDER_COMMAND_ACTION	Print
3	PROCESS_WEB_ORDER_COMMAND_ACTION	Process web order via WebOrderProcessor service.
4	RETRY_WEB_ORDER_COMMAND_ACTION	Retry web order processing via WebOrderProcessor service.

### <sup>2</sup> Status Values

Value	Gateway Constant Name	Description
0	AVAILABLE_ORDER_COMMAND_STATUS	Order Command is available to be processed
1	PROCESSING_ORDERCOMMAND_STATUS	Order Command is currently being processed
2	ERROR_ORDERCOMMAND_STATUS	An error occurred while processing the Order Command

## 15.49 OrderCommunications

The Order Communication table stores all of the guest communications that will be sent either for Order Confirmation or Fulfillment.

### Columns

Column	Type	Allow Nulls	Description
OrderCommunicationID	Int	N	Primary key, always unique
MechanismType <sup>1</sup>	Int	N	The mechanism that will be used to send this communication.
RecipientType <sup>2</sup>	Int	N	Which recipient in the order will receive this communication.
EmailTemplateID	Int	Y	If this is an email (MechanismType = 0), the ID of the record in WebTemplates that defines the template for this email.
SMSTemplateID	Int	Y	If this is an SMS message (MechanismType = 1), the ID of the record in SMSTemplates that defines the template for this SMS.
Scope <sup>3</sup>	Int	N	Scope of this communication
ProductAttachments <sup>4</sup>	Int	N	How message attachments will be handled.
CommunicationGroupID	UniqueIdentifier	N	A unique identifier that links a set of OrderCommunication records with a given Delivery Method, Category Group, Customer Category, or Customer. This unique identifier is also stored on the associated Delivery Method, Category Group, Customer Category, or Customer record.

### Indexes

Name	Kind	Columns	Purpose
PKOrderCommunicationID	P	OrderCommunicationID	Primary Key

#### <sup>1</sup> MechanismType Values

Value	Gateway Constant Name	Description
0		Email
1		SMS

#### <sup>2</sup> RecipientType Values

Value	Gateway Constant Name	Description
0		Order Contact
1		Ship-To Contact
2		Product Contact (ticket holder, etc.)

#### <sup>3</sup> Scope Values

Value	Gateway Constant Name	Description
0		Order - communication sent about an order (eg., an order confirmation email)
1		Product - communication sent about a product (eg., a PDF ticket)

#### <sup>4</sup> ProductAttachments Values

Value	Gateway Constant Name	Description
0		Will not be sent
1		Attachments will be combined into the same file
2		Attachments will be separate files attached to the same message

## 15.50 OrderDetails

After a ticket has been issued, Order Entry stores the individual ticket information in this table.

### Columns

Column	Type	Allow Nulls	Description
OrderDetailID	Int	N	Primary key, always unique.
OrderLineID	Int	N	Foreign key to OrderLines.OrderLineID.
PLU	Char(20)	N	An alphanumeric code uniquely identifying a particular ticket or retail item.
ItemQty	Int	Y	The quantity of tickets in this detail record (always 1).
ItemIssuedQty	Int	Y	The quantity of tickets that have been issued (always 1).
ItemPrice	Money	N	The price of the ticket.
ItemTax	Money	N	Tax amount for this ticket.
HasRateAdjustment	Bit	N	If TRUE, indicates that this ticket has a Sales Program applied to the price.
EventID	Int	Y	Foreign key to RMEvents.EventID.
ReservationID	Int	Y	Foreign key to RMReservations.ReservationID.
RentalSerialID	Int	Y	Foreign key to RMRentalSerials.RentalSerialID.
HasRentalInventory	Bit	Y	If TRUE, indicates that the ticket has a rental Capacity.
ResourceID	Int	Y	Foreign key to RMResources.ResourceID.
NodeNumber	Int	N	Node number for this ticket.
TransactionNumber	Int	N	Transaction number this ticket belongs to.
TransactionDateTime	DateTime	N	Date and time of the transaction to which this ticket belongs.
AgentNumber	Int	N	Agent number for this ticket.
TicketInTrans	Int	Y	Ticket number in this transaction, used for disbursements.
DiscountID	Int	Y	Discount ID for this ticket.
DiscountAmount	Money	Y	Amount of discount for this ticket.
AccessCode	Int	Y	Access code associated with this ticket.
VisualID	Char(40)	Y	The unique Visual ID associated with this ticket.
SerialNumber	Int	Y	The serial number associated with this ticket.
TicketDate	DateTime	N	Ticket date if using date-specific tickets, otherwise this will be 01/01/1900.
OrderTransactionID	Int	Y	Foreign key to OrderTransactions.OrderTransactionID
Company	Int	Y	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
Category	Int	Y	The transaction account category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SubCat	Int	Y	The transaction account sub-category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
ExpirationDate	DateTime	Y	Store expiration date of a Date Range Ticket
SuperTicketVisualID	VarChar(40)	Y	The VisualID of the super ticket associated with the ticket.
ContactID	Integer	Y	Foreign Key, references CustContacts.CustContactID
DetailID	Int	Y	Foreign key reference to the table referenced by DetailType below
DetailType <sup>1</sup>	Int	Y	Indicates that type of information contained in the detail record <sup>1</sup>
SalesProgramID	Integer	Y	Foreign key to SalesPrograms.SalesProgramID
Points	Integer	Y	The point value journalized
CapacityID	Integer	Yes	The unique ID of the related RMCapacity record
GuestFirstName	Varchar(30)	Y	First name of the guest for the ticket on this detail
GuestLastName	Varchar(30)	Y	Last name of the guest for the ticket on this detail
EndOfLifeDate	DateTime	Y	End of life date

### Indexes

Name	Kind	Columns	Purpose
PKOrderDetailsOrderDetailID	P	OrderDetailID	Primary Key.
IXOrderDetailsOrderLineID	F	OrderLineID	Used to select the detail records for a particular order line.
IXOrderDetailsOrderTransID		OrderTransactionID	Search tickets by the OrderTransactionID
IXOrderDetailsVisualIDPLU		VisualID, PLU	Used when querying data to void an order entry transaction.
IXOrderDetailsBatchPrintReport		OrderTransactionID, PLU, OrderLineID	Batch Print Report
IXOrderDetailsTransDate		TransactionDatetime	Used when searching orders by IssueDate in Order Manager

### <sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
1	ID_ITEM_ORDERDETAIL	The order detail contains draft information
2	ID_PAYMENT_ORDERDETAIL	The order detail contains payment information
3	ID_NOTE_ORDERDETAIL	The order detail contains notes
4	ID_RETURN_PAYMENT_ORDERDETAIL	The order detail contains return payment information
5	RESERVED	Not used
6	ID_TAX_ORDERDETAIL	The order detail contains tax information
7	ID_VOIDTAX_ORDERDETAIL	The order detail contains voided tax information
8	ID_PASS_ORDERDETAIL	The order detail contains pass information
9	ID_TOTAL_TAX_ORDERDETAIL	The order detail contains total tax information
10	ID_PACKAGE_ORDERDETAIL	The order detail contains package information

12	ID_PACKAGE_DETAIL_ORDERDETAIL	The order detail contains package detail information
13	ID_FEE_ORDERDETAIL	The order detail contains fee information
14	ID_TRANSPORTATION_ORDERDETAIL	The order detail contains transportation information
15	ID_DONATION_ORDERDETAIL	The order detail contains donation information
	ID_JOINTMEMBERADDON_ORDERDETAIL	The order detail contains joint member add-on information

### 15.51 OrderDiscounts

Stores discount information related to orders.

#### Columns

Column	Type	Allow Nulls	Description
OrderDiscountID	Int	N	Primary key, always unique.
OrderID	Int	N	Foreign key to Orders.OrderID
DiscountID	Int	N	Foreign key to Discounts.DiscountID
ApplyCount	Int	Y	The number of number of times the discount ID has been applied successfully
SupervisorID	Int	Y	ID of the supervisor that approved the discount. Used when supervisor approval is configured on a discount. FK reference to GxUsers.UserID.
RequiredPassID	Int	Y	ID of the pass used for a pass-required discount. FK reference to Passes.PassNo.

#### Indexes

Name	Kind	Columns	Purpose
PKOrderDiscountID	P	OrderDiscountID	Primary Key.
IXOrderDiscountsOrderID	IX	OrderID	Foreign key to Order

## 15.52 OrderDrafts

Draft information for an Order Entry bank card payment

### Columns

Column	Type	Allow Nulls	Description
OrderDraftID	Int	N	Primary key, always unique.
OrderTransactionID	int	N	Unique Foreign key to OrderTransactions.OrderTransactionID
AuthCode	VarChar(128)	Y	Approval Code (returned from host). <sup>1</sup>
TransID	Char(15)	Y	Transaction ID (returned from host). <sup>1</sup>
Validation	Char(4)	Y	Validation Code (returned from host). <sup>1</sup>
PSI	Char(1)	Y	Payment Service Indicator (returned from host). <sup>1</sup>
FOP	Int	Y	The form of payment used to tender payment
CardNo	Varchar(50)	Y	The credit card number used to tender payment (to debit / gift card account number)
JnlCode	Int	Y	The journal code for the draft record <sup>2</sup>
GxKeyID	Int	Y	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.
HostTraceNumber	Char(8)	Y	Trace number returned from host for some credit card protocols
TransID2	Varchar(20)	Yes	Contains transaction ID for Protocol Host. Where TPT is system prefix, contains the unique trace reference that Galaxy generates to identify the transaction, and format is TPTYYYYMMDDHHMMSSNNN.
HostTransDateTime	DateTime	Yes	Date and time of the transaction
CardType	Char(1)	Yes	Identifies the card type used <sup>3</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderDraftsOrderDraftID	P	OrderDraftID	Primary Key.
IXOrderDraftsLastUpdate		LastUpdate	Optional Index to improve performance of DB Update Process in DeleteCCInfoUtility.
IXOrderDraftsOrderTranIDAuthCode		OrderTransactionID, AuthCode	Index used in Payment Contract report.

<sup>1</sup> Note: Depending upon the credit card protocol being used, the format of these fields can vary. For certain protocols, some of these fields may always be empty or zero.

<sup>2</sup> JnlCode Values

Value	Gateway Constant Name	Description
35	DRAFT_REC	Normal Draft record
69	STRATUS_ACTIVATION_REC	Stratus activation authorization
70	STRATUS_RECHARGE_REC	Stratus recharge authorization

<sup>3</sup> CardType Values

Value	Description
V	Visa
M	Mastercard
A	American Express
J	JCB
D	Diners
C	Cup
E	EZ Link

**15.53 OrderItemDetails**

Contains information for journalized (issued) retail items for an order.

**Columns**

Column	Type	Allow Nulls	Description
OrderItemDetailID	Int	N	Primary key, always unique.
OrderLineID	Int	Y	Foreign key to OrderLines.OrderLineID. The unique id of the order line to which this detail refers.
PLU	Char(20)	Y	The PLU number for this retail item.
ItemQty	Int	Y	The quantity of items in this detail record
ItemPrice	Money	Y	The extended price of the item (qty times unit price)
ItemTax	Money	Y	The tax amount for the item
DiscountAmount	Money	Y	The discount amount for the item
OrderTransactionID	Int	Y	Foreign key to OrderTransactions.OrderTransactionID
Company	Int	Y	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
Category	Int	Y	The transaction account category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SubCat	Int	Y	The transaction account sub-category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SalesProgramID	Integer	Y	Foreign key to SalesPrograms.SalesProgramID

**Indexes**

Name	Kind	Columns	Purpose
PKOrderitmDtlOrderItemDtlID	P	OrderItemDetailID	Primary Key.
IXOrderitmDtlOrderLineID		OrderLineID	Search items by OrderLineID
IXOrderitmDtlOrderTransID		OrderTransactionID	Search items by OrderTransactionID

### 15.54 OrderLineDiscounts

The OrderLineDiscounts table is a detail table relating to OrderLines. A single order line can have one to many records in this table, each record representing a discount applied to that order line's ticket(s) or item(s). Entries into this table represent the total discounts applied to all tickets and items associated with their associated order line.

#### Columns

Column	Type	Allow Nulls	Description
OrderLineDiscountID	Int	N	Primary key, always unique
OrderLineID	Int	N	Foreign key reference to OrderLines.OrderLineID
DiscountAmount	Money	N	Amount of the discount
DiscountKind	Int	N	Kind of discount <sup>1</sup>
RequirementCount	Int	N	Number of requirement entries (out of the total number of tickets) for the given order line
DiscountID	Int	N	Foreign key to the applied discount, depends on DiscountType. Could be a POS Discount ID, Sales Program ID. If value is zero, a price override is in effect.
CanDiscountRequirement	Bit	Y	Can be set to 1, when order line discount record is a requirement. When set to 1, the requirement can be discounted.
OrderLineDiscountReplacementID	Int	Y	FK reference to OrderLineDiscountReplacements. OrderLineDiscountReplacementID column
DiscountOccurrence	Int	Y	Occurrence of a given POS DiscountID for the order. Value in this column is used to determine how many times a given POS discount is applied to an order

#### Indexes

Name	Kind	Columns	Purpose
PKOrderLineDiscountID	P	OrderLineDiscountID	Primary Key
IXOrderLineDiscountsOrdLineID		OrderLineID	Search discounts by OrderLineID

#### <sup>1</sup> DiscountKind Values

Value	Gateway Constant Name	Description
0	DISCOUNT_KIND_PRICE_OVERRIDE	User overrode the price of the ticket or item
1	DISCOUNT_KIND_SALES_PROGRAM	Sales program changed the price of the ticket or item
2	DISCOUNT_KIND_COUPON_DISCOUNT	A POS/Coupon discount was applied to the ticket or item
3	DISCOUNT_KIND_ITEM_DISCOUNT	Item discount was applied
4	DISCOUNT_KIND_LINE_ITEM_DISCOUNT	Line item discount was applied

### 15.55 OrderLineDiscountReplacements

OrderLineDiscountReplacements table is a detail table for OrderLineDiscounts table. OrderLineDiscounts table contains discount breakup for an OrderLine. OrderLineDiscountReplacements table stores replacements applied to a discount breakup of an OrderLine. Replacements are applied to an OrderLine when a discount with replacements is applied to an OrderLine in Order Entry.

#### Columns

Column	Type	Allow Nulls	Description
OrderLineDiscountReplacementID	Int	N	PrimaryKey, always unique
Company	Int	Y	Company used for the replacement ticket, if that gets activated when the discount is applied
Category	Int	Y	Category used for the replacement ticket, if that gets activated when the discount is applied
SubCategory	Int	Y	SubCategory used for the replacement ticket, if that gets activated when the discount is applied
Printer	Int	Y	Printer used for the replacement ticket, if that gets activated when the discount is applied
TicketSet	Int	Y	TicketSet used for the replacement ticket, if that gets activated when the discount is applied
DisbursementID	Int	Y	DisbursementID used for the replacement ticket, if that gets activated when the discount is applied
AccessCode	Int	Y	AccessCode used for the replacement ticket, if that gets activated when the discount is applied

#### Indexes

Name	Kind	Columns	Purpose
PKOrderLineDiscReplacementID	P	OrderLineDiscountReplacementID	Primary Key

## 15.56 OrderLines

This table stores the contents of an Order, including information on tickets, retail items, notes, payments, and taxes.

### Columns

Column	Type	Allow Nulls	Description
OrderLineID	Int	N	Primary key, always unique.
Description	VarChar(100)	Y	The item description, payment FOP description, or tax name, depending on the record's detail type.
OrderID	Int	N	Foreign key to Orders.OrderID.
InvoiceID	Int	Y	Foreign key to ARInvoices.InvoiceID. This value will be zero or NULL if this order line has not been invoiced.
NoteID	Int	Y	Foreign key to Notes.NoteID.
DetailTableID	Int	Y	Identifies where more information on this record can be found, depending on the value of the DetailType <sup>1</sup> .
DetailType	Int	N	A numeric code identifying the type of OrderLine. <sup>1</sup>
Quantity	Int	Y	Quantity for this line.
IssuedQuantity	Int	Y	Number of issued items for this line.
PLU	char(20)	Y	An alphanumeric code uniquely identifying a particular ticket or retail item..
Amount	Money	Y	The unit price for this line. This is the price from the items table <b>with all taxes removed</b> (this is true even for tax-included tickets or items). The discount amount has <b>not</b> been removed from this price.
			When the line's DetailType indicates that this is a "tax" line, this column will contain the total tax amount for previously issued order lines.
Total	Money	Y	Total amount for all items in this line, using the calculation:  ((Amount + TaxAmount - Discount Amount) * Quantity).
DiscountAmount	Money	Y	Discount amount per ticket/item for this line. To determine what sort of discount is contained here, see the PriceBasis column. <sup>2</sup>
EventID	Int	Y	Foreign key to RMEvents.EventID.
TicketDate	DateTime	Y	Date for Date Specific Tickets.
ReservationID	Int	Y	Foreign key to RMReservations.ReservationID.
RentalSerialID	Int	Y	Foreign key to RentalSerial.RentalSerialUniqueID.
HasRentalInventory	Bit	N	If TRUE, indicates that this line has a Rental Inventory.
ResourceID	Int	Y	Foreign key to RMResources.ResourceID.
LeadGuestID	Int	Y	Foreign key to OEGuest.OEGuestID, representing the Lead Guest for this line.
GuestID	Int	Y	Foreign key to OEGuests.OEGuestID, representing the guest for this line.
RoomID	Int	Y	Number of the Room for this line.
LineNbr	Int	Y	OrderLineID used to set the display sequence of order lines within each order. <b>Should only be altered by the system.</b>
CreditMemoid	Int	Y	Foreign key to ARCreditMemo.CreditMemoid, if a Credit Memo has been applied to this line. Zero or NULL if one has not.
DiscountBarcode	Char(40)	Y	A barcode required by the ticket or item, in order for it to be sold.
BatchDetailsID	Int	Y	Foreign key to the OEBatchPrintHeader.BatchHeaderID, if this order line has been assigned to a Batch Print batch. Otherwise, this contains zero or NULL.
TaxAmount	Money	Y	This is the amount of tax calculated on each unit of the ticket or item, whether it is tax-included or tax-excluded. However, any transactional taxes that may be applied to this ticket or item will not be included here. Transactional taxes will appear in separate "tax" order lines.
			When the line's DetailType indicates that this is a "tax" line (DetailType = 6), this column will contain the total transactional tax amount for previously invoiced order lines.
PriceBasis	Int	Y	The type of value in the Amount and Discount Amount fields. <sup>2</sup>
SystemSplitLine	bit	Y	If true (1), indicates that the Sales Program split the order line from the original entry. The Sales Program can merge/split an OrderLine where this field is set to true.
QtyInBatch	Int	Y	The quantity of tickets from this orderline is assigned for batch printing.
ExpirationDate	DateTime	Y	Store expiration date of a Date Range Ticket
PassID	Int	Y	Foreign key to Passes.PassNo
			Used for DetailType of 8 (Pass Record). Specifies that the Pass with given ID is linked to the OrderLine
SalesChannelDetailID	Int	Y	FK reference to the SalesChannelDetails.SalesChannelDetailID
			ID of the "Sales Sub Category Detail" used when sending the confirmation e-mails to get the marketing message IDs
ContactID	Int	Y	Foreign Key, references CustContacts.CustContactID
OrderLineGiftDetailID	Int	Y	Foreign key to OrderLineGiftDetails table.
SalesProgramID	Integer	Y	Foreign key to SalesPrograms.SalesProgramID
InvoiceLineID	Integer	Y	FK reference to ARInvoiceLines.InvoiceLineID
CreateDate	DateTime	Y	Create date of the order line
DisbursementID	Int	Y	FK reference to Disbursements.DisbursementID column
PaymentContractID	Int	Y	FK to PaymentContracts.PaymentContractID, if this Order Line is in a Contract. Otherwise, zero or null.
PaymentPlanID	Int	Y	FK reference to PaymentPlans.PaymentPlanID . Unique ID of the Payment Plan associated with the order line.
GroupID	Int	Y	UniqueID of grouping order line
PackageDetailID	Int	Y	Foreign key reference to the PackageDetails.PackageDetailID
Points	Integer	Y	The point value of the PLU that is associated to the order line
CapacityID	Integer	Yes	The unique ID of the related RMCapacity record
HasDiscount	Bit	Y	Set to 1, when a POS discount (with non-zero discount amount) is applied to this OrderLine
VisualID	Varchar(40)	Y	VisualID of the pass. Used for Pass order lines (DetailType of 8). There can be only one Pass per OrderLine.  This column is currently only used by the Web store.
SalesSubCategoryID	Money	Y	SalesChannelDetailID on the Sales Category of item in this OrderLine
DeliveryMethodGroupID	Int	Y	Foreign Key to DeliveryMethodGroups

ShippingPLU	Int	Y	Whether the OrderLine is a shipping PLU or not (0: No; 1: Yes)
EventTicketDate	DateTime	Y	Date for capacity managed ticket
NameOnPass	Varchar(150)	Y	For pass renewal
NewExpPass	DateTime	Y	For pass renewal
WSDetailType	Int	Y	Used in Associated Tickets processing
BaseID	Int	Y	Used to group related orderlines together = OrderLineID of the primary orderline for the group
BusID	Int	Y	Link to Order Line Bus Tickets
RedeemedValue	Float	Y	Loyalty 'points' redeemed for this orderline
TransID2	Varchar(20)	Y	Transaction ID from protocol host system.
RSEventSeatMapID	Int	Y	Link to RSEventSeatMaps, is populated when event is selected for event tickets
ExpirationOverrideDate	DateTime	Y	This date is set to a non-null value if the user overrode the expiration date on a pass renewal when it was added to the order. The expiration date for pass renewals on an order is updated only when the renewal is issued, and this value is needed to preserve expiration date modifications by the user made before issuance.
EventTicketEndDate	DateTime	Y	The event's end date and time for event ticket
EventName	Varchar(250)	Y	The event ticket's event name
IsPassRequired	Bit	Y	Indicates that the orderline item is pass required.
PendingLoyaltyPoints	Float	Y	Loyalty points that will be earned when the order line is issued. This value is transferred to IssuedLoyaltyPoints when the order line is issued.
IssuedLoyaltyPoints	Float	Y	Loyalty points that were earned when the order line was issued. This value is transferred from PendingLoyaltyPoints when an order is issued.
LoyaltyAccountNo	Varchar(40)	Y	Account number of the Loyalty Program associated with the Order
LoyaltyProgramID	Integer	Y	Foreign key to LoyaltyPrograms.LoyaltyProgramID
GiftAidAmount	Money	Y	GiftAid amount for this orderline
GiftAidType	Int	Y	Type of GiftAid applied to the orderline <sup>3</sup>
HasGuests	Bit	Y	Value is true if the orderline has OEGuests associated with it.
IsNotPrinted	Bit	Y	A value of 1 in this field indicates that tickets or passes for this order line have been created but not printed. This happens for SIAE-enabled systems if a payment is made before the order is issued. This is set back to 0 when the order is issued.
ExternalDiscountID	VarChar(20)	Y	Promotion code used to get reduced pricing for this ticket.
ExternalDiscountName	VarChar(80)	Y	Name of the promotion used to get reduced pricing for this ticket
NodeNo	Int	Y	Node that created the orderline
PackageVisualID	VarChar(40)	Y	When the OrderLine is a sale of deferred details from a Package, this field will refer to the Visual ID of the original package.
SaleSupervisorID	Int	Y	ID of the Supervisor that approved the sale, or who approved a downgrade refund. FK reference to GxUsers.UserID.
EntitlementAddOnVisualID	nvarchar(40)	Y	Contains the VisualID of the entitlement that this item will be added to.
IsAssociatedTicket	Bit	Y	This field is used to indicate that the order line was added to the order because of the associated ticket ratio.
ResellerExternalPrice	Money	Y	The reseller price that is calculated based on price schedules configuration.  This column is only used by the reseller web store.
ResellerItemPrice	Money	Y	The reseller PLU price that is calculated based on price schedules configuration.  This column is only used by the reseller web store.
PriceProgramID	Int	Y	If this item is configured to use the "Visit Date" price method, this column holds the PriceProgramID of the active price program for this item for this order's visit date.
UpgradeValue	Money	Y	Value is retrieved from static pricing or the pricing plugin when the orderline record is saved.
PkgInstanceDetailID	Int	Y	Foreign key reference to the PkgInstanceDetails table.
DeferredEntitlement	Bit	Y	Indicator that this order line is a deferred entitlement.
PriceToken	Varchar(60)	Y	An identifier used to reference the associated external price data for future pricing requests.
HasModifiedUpgradeValue	Bit	Y	If this is a return order line for an upgrade or an order line for a pass upgrade, indicates whether the upgrade value was modified by the user.
ModifiedUpgradeValue	Money	Y	If this is a return order line for an upgrade or an order line for a pass upgrade, and if the user modified the upgrade value, contains the new upgrade value.
PreventRecalculation	Bit	Y	Indicates whether the data on this order line can be recalculated.

**Indexes**

Name	Kind	Columns	Purpose
PKOrderLinesOrderLineID	P	OrderLineID	Primary Key.
IXOrderLinesOrderLineIDRecVer		OrderLineID, RecordVersion	Used to speed-up the Auto Batch Print process.
IXOrderLinesOrderIDLineNbr		OrderLineID, LineNbr	Used to speed-up the Auto Batch Print process.
IXOrderLinesPassIDDetailType		PassID, DetailType	Used by query to get OrderID for the given PassID and DetailType
IXOrderLinesEventID		EventID	Speed up Event Activity query in Event Availability
IXOrderLinesOrderIDDetailType		OrderID, DetailType	Index is used by SelectByOrderID4SalesByEventReport query for the Detail Sales by Event Report in Agency Reports
IXOrderLinesOrderSummary		OrderID, DetailType, Quantity, IssuedQuantity	Order Summary Report
IXOrderLinesGuestID		GuestID	Used to improve performance of OrderManager
IXOrderLinesLeadGuestID		LeadGuestID	Used to improve performance of OrderManager
IXOrderLinesReservationID		ReservationID	Used by queries selecting order lines by ReservationID
IXOrderLinesGroupID		GroupID	Alternate key for GroupID column
IXOrderLinesPackageDetailID		PackageDetailID	Alternate key for PackageDetailID column
IXOrderLinesSalesProgramID		SalesProgramID	Used to improve performance when deleting sales programs

<sup>1</sup> DetailType Values

Value	Description	Detail Table ID

1	Ticket/item detail record	Foreign key to OrderDetails.OrderDetailID.
2	Payment detail record	Foreign key to OrderPayments.OrderPaymentID.
3	Note/memo record	Foreign key to Notes.NoteID.
4	Return payment record	Foreign key to OrderPayments.OrderPaymentID.
5	Reserved for use by invoice records. <i>This value is not currently used by order lines.</i>	
6	Tax record	The Tax ID, currently 1 thru 8.
7	Voided tax	
8	Pass detail record	Foreign key to Passes.PassID.
9	Detail Type 9 is used to store the total amount of tax on the order per TaxID  • Amount is used to store the total non-transactional tax amount • TaxAmount is used to store the total transactional tax amount • Total is the sum of Amount & TaxAmount	The Tax ID, currently 1 thru 8.
10	Package record	
11	Package detail record	Foreign key reference to the PackageDetails.PackageDetailID
12	Fee detail record	
13	Transportation detail record	
14	Donation detail record	
15	Joint Member detail record	
16	Upgrade Return detail record	
17	Package Upgrade Return detail record	
18	Package Detail Upgrade Return detail record	
19	Partial Package Upgrade Return detail record	
20	Renewal Credit Return detail record	
21	Package Renewal Credit Return detail record	
22	Package Detail Renewal Credit Return detail record	
23	Partial Package Renewal Credit Return detail record	

**<sup>2</sup> PriceBasis Values**

Value	Description
0	Normal price (no discount)
1	Sales Program has been applied
2	Price Override has been applied
10	Normal price (no discount) - Package Item Price is locked
11	Sales Program has been applied - Package Item Price is locked
12	Price Override has been applied - Package Item Price is locked

**<sup>3</sup> GiftAid Type Values**

Value	Gateway Constant Name	Description
0	gaNone	No GiftAid allowed for the item
1	gaFixedAmount	The Gift Aid is a fixed amount from the item definition.
2	gaFull	Full ticket price can be used as GiftAid
5	gaPercentage	Gift Aid is calculated as a percentage of the item price.

### 15.57 OrderLineGiftDetails

This table stores the Gift information for an OrderLine. This is populated if OrderLine item has RegisterGift option turned on.

#### Columns

Column	Type	Allow Nulls	Description
OrderLineGiftDetailID	Int	N	Primary key, always unique.
OrderID	Int	N	Foreign key to Orders.OrderID
OrderLineID	Int	N	Foreign key to OrderLines.OrderLineID
CampaignID	Int	N	Foreign key to Campaigns.CampaignID
FundID	Int	N	Foreign key to Funds.FundID
AppealID	Int	N	Foreign key to Appeals.AppealID
SolicitationID	Int	N	Foreign key to Solicitations.SolicitationID

#### Indexes

Name	Kind	Columns	Purpose
PKOrderLineGiftDetailID	P	OrderLineGiftDetailID	Primary Key.

## 15.58 OrderLineJointMembers

The OrderLineJointMembers table contains information about joint members that are associated with joint member passes contained in orders. Currently, this table is used to keep track of joint member information during a joint member pass split operation.

Name	Type	Allow Nulls	Description
JointMemberID	Integer	N	Primary key, always unique
PassNo	Integer	N	Pass the joint member is associated to. Foreign key to Passes.PassNo.
ContactID	Integer	N	Contact containing demographic information about this joint member. Foreign key to CustContacts.CustContactID.
VisualID	VarChar(40)	Y	Visual ID for the membership card associated with this member.
PictureID	Integer	Y	Picture associated with this member. Foreign key to Pictures.PictureID.
PreviousID	Integer	Y	The JointMemberID of the previous joint member record in the chain. When a joint membership is renewed, reissued/reprinted, or upgraded, this points to the previous joint member record for the same member.
MemberType	Integer	N	Indicates the type of joint member. <sup>1</sup>
PrimaryMember	Bit	Y	Indicates if the joint member is the primary member for the membership.
PLU	Char(20)	Y	The PLU of the item that the member was sold under. This field is only used for members that are sold separately from the core membership (such as add-on members).
Status	Integer	Y	Status of the joint member. This is only set if the member status is different than the status of the membership. <sup>2</sup>
OrderLineID	Integer	Y	Foreign key to OrderLines.OrderLineID
RelationshipTypeID	Integer	Y	Foreign key to RelationshipType.RelationshipTypeID
ReciprocalRelationshipTypeID	Integer	Y	Foreign key to RelationshipType.RelationshipTypeID
Splitting	Bit	N	This joint member record is involved in a joint member pass split operation.
ExistingContactAddedForSplit	Bit	N	
JointMemberID	Integer	Y	Foreign key to JointMembers.JointMemberID
ListType	Integer	N	Indicates the type of operation in which the joint member record is involved. <sup>3</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderLineJointMemberID	P	OrderLineJointMemberID	Primary key
IXOrderLineJMOrderLineID		OrderLineID	Speed up queries to load order line joint member records for a given order line.

### <sup>1</sup> MemberType Values

Value	Gateway Constant Name	Description
0	MEMBER_TYPE_STANDARD	Normal joint member added as part of the membership.
1	MEMBER_TYPE_ADD_ON	Add-on member sold separately from the core membership.

### <sup>2</sup> Status Values

| Value | Gateway Constant Name | Description | -----|-----|-----|-----| 0 | JOINT\_MEMBER\_VALID | Valid | | 1 | JOINT\_MEMBER\_VOIDED | Voided | | 2 | JOINT\_MEMBER\_RETURNED | Returned | | 3 | JOINT\_MEMBER\_REMOVED | Removed | | 4 | JOINT\_MEMBER\_SPLITOFF | Split Off | | 5 | JOINT\_MEMBER\_REPLACE | Replaced | ###<sup>3</sup> ListType Values {-}

Value	Gateway Constant Name	Description
0	INVALID_LIST_TYPE	This is an invalid value for a joint member list type.
1	RENEWING_SPLIT_LIST_TYPE	Indicates that the joint member is involved in a joint membership split operation.

### 15.59 OrderLineSeats

The OrderLineSeats table links one to many reserved seats to an order line.

#### Columns

Column	Type	Allow Nulls	Description
OrderLineSeatID	Int	N	Primary key, always unique.
OrderLineID	Int	N	FK to OrderLines table
RSEventSeatID	Int	N	FK to RSEventSeats table
SectionName	Char(10)	N	Name of the section
RowName	Char(10)	N	Name of the row
SeatName	Char(10)	N	Name of the seat
RSSeatID	Int	N	FK to RSSeats table
SessionGUID	uniqueidentifier	Y	Holds the session that was used to reserve the seat recorded in the OrderLineSeats table.

#### Indexes

Name	Kind	Columns	Purpose
PKOrderLineSeatID	P	OrderLineSeatID	Primary Key.

## 15.60 OrderLineTaxAmounts

This table stores the individual tax amounts for a particular OrderLine.

### Columns

Column	Type	Allow Nulls	Description
OrderLineTaxAmountID	Int	N	Primary Key, always unique
OrderLineID	Int	N	Foreign key reference to OrderLines.OrderLineID
TaxID	Int	N	Tax identification
TaxTableID	Int	Y	
DisburseIndex	Int	Y	If the tax amount is for a disbursement, this value represents the SequenceNo of the disbursement detail that the tax amount is for. The current values are 1-10, or 0 if it is not for a disbursement
UnitAmount	Money	Y	For transactional taxes, this represents the individual price of the item being charge tax. Otherwise, this is the actual amount of tax being charged.
ReducePrice	Bit	Y	If set to true, the tax amount (UnitAmount) was included in the original price of the ticket or item. In other words, the ticket or item has the tax included flag turned on.
Method	Int	Y	0: Item base tax 1: Transactional tax
Status	Int	Y	Indicates whether or not the tax amount is applicable <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderLineTaxAmountID	P	OrderLineTaxAmountID	Primary Key.
IXOrdLineTaxAmountsOrderLineID	IX	OrderLineID	Used when retrieving tax data by OrderLineID

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	TAX_AMOUNT_NOT_UPDATED	
1	TAX_AMOUNT_APPLICABLE	
2	TAX_AMOUNT_NOT_APPLICABLE	

### 15.61 OrderLineTransportationTickets

This table stores the contents of a transportation order. It includes origin, destination and travel dates.

#### Columns

Column	Type	Allow Nulls	Description
OrderLineTransTktID	Int	N	Primary key, always unique
OrderID	Int	N	Link to Order table
OrderLineID	Int	N	Link to Order Line table
Origin	Int	Y	Information about the origin city
Destin	Int	Y	Information about the destination city
Description	VarChar(100)	Y	Description
TripMode	Int	Y	Trip mode, one way/ round trip
OWRT	Char(2)	Y	One way or Round Trip
DepartureDate	DateTime	Y	Starting date of travel
ReturnDate	DateTime	Y	Return date of travel
TariffID	Int	Y	Id of the Tariff Record
TariffName	Char(24)	Y	Name of the Tariff
Endorsement	Char(24)	Y	
PassengerName	VarChar(30)	Y	Passenger Name
HasTariffMiles	Bit	Y	Does this have Tariff miles?
TariffMiles	Int	Y	Tariff Miles
FareMiles	Int	Y	Fare Miles
SurchargeAmount	Float	Y	Any surcharge amounts
FareID	Int	Y	Id to the Fare record
Restrictions	VarChar(64)	Y	Restriction codes
TicketTypeID	Int	Y	Link to TransportationTicketTypes table
OutboundItinID	Int	Y	Link to Itineraries table
InBoundItinID	Int	Y	Link to Itineraries table
TariffCarrier	Char(4)	Y	Carrier for the tariff definition
IsCommuter	Bit	Y	Is this a commuter ticket
Nrides	Int	Y	How many rides is the user allowed
OWLlimit	Int	Y	From Fares table record. Number of days a one way ticket is valid.
RTLlimit	Int	Y	From Fares table record. Number of days a round trip ticket is valid.

#### Indexes

Name	Kind	Columns	Purpose
PKOLTransTktsOrdrLneTransTktID	P	OrderLineTransTktID	Primary Key.

## 15.62 OrderLineUpgrades

This table stores information about which user performed an upgrade if that user edited details about the transaction that will be needed during issuance for audit logging. For instance, if a user edits the upgrade value of a ticket, a row will be created in this table to indicate: a) the date/time/user/node/agency that performed the upgrade, and b) the original and new upgrade values. Later, when the tickets are issued, that information is used to create audit logs indicating the original user that performed the upgrade.

Additionally, this table stores information about downgrades of items that are configured to maintain price and upgrade value of original entitlement in downgrade.

Note: This table does not contain a record for every upgrade performed through Order Entry. Currently, it only contains a record for upgrades where the upgrade value was edited by the user, and for downgrades of items that are configured to maintain price and upgrade value of original entitlement in downgrade.

### Columns

Column	Type	Allow Nulls	Description
OrderLineUpgradeID	Integer	N	Primary key, always unique.
OrderLineID	Integer	N	Foreign key to OrderLines.OrderLineID, the order line containing the upgrade
UserID	Integer	Y	The user that executed the Ticket Upgrade function and selected this upgrade. This is not the user that issued the order line (if issued).
NodeNo	Integer	Y	The node that executed the Ticket Upgrade function and selected this upgrade.
AgencyNo	Integer	Y	The agency in which the node was that executed the Ticket Upgrade function and selected this upgrade.
UpgradeDate	DateTime	Y	The date/time at which the Ticket Upgrade function was executed. This is not the date/time the tickets were issued (if issued).
OriginalUpgradeValue	Money	Y	The original upgrade value of the ticket/pass.
EditedUpgradeValue	Money	Y	The edited upgrade value of the ticket/pass.
UpgradeType	Integer	Y	A numeric code indicating the type of upgrade performed. <sup>1</sup>
ConfiguredUpgradePrice	Money	Y	The normal price of the upgrade.
AppliedUpgradePrice	Money	Y	The upgrade price charged to the guest.

### Indexes

Name	Kind	Columns	Purpose
PKOrderLineUpgradeID	P	OrderLineUpgradeID	Primary Key.
IXOrderLineUpgradesOrderLineID	IX	OrderLineID	Used when retrieving data by OrderLineID

### <sup>1</sup> UpgradeType Values

Value	Description
NULL/0	Edited upgrade value
1	Downgrade without refund

## 15.63 OrderLineUpsells

This table stores the upsell information for an orderline. Upsell information is gathered at the time an item is added to a transaction and must be stored for journalization at the time of issue.

### Columns

Column	Type	Allow Nulls	Description
OrderLineUpsellID	Integer	N	Primary key, always unique.
OrderLineID	Integer	N	Foreign key to OrderLines.OrderLineID
PLU	VarChar(20)	N	The original PLU that was selected. This field will be blank if the user was presented with options for upsell, but no upsell options were chosen.
PriceDifference	Money	N	The difference in price between the original PLU selected, and the upsell item chosen. This is the full price of the item for Add-On upsell options.
UpsellType	Integer	N	Type of upsell. This determines how the upsell option is presented and selected. <sup>1</sup>
UserID	Integer	N	ID of the user that performed the upsell.
SalesChannelType	Integer	N	The sales channel type indicates where the upsell was performed (POS, OE, kiosk, Web). <sup>2</sup>
UpsellStatus	Integer	Y	Indicates if the upsell was selected or cancelled. <sup>3</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderLineUpsellID	P	OrderLineUpsellID	Primary Key.
IXOrderLineUpsellsOrderLineID	IX	OrderLineID	Used when retrieving data by OrderLineID

<sup>1</sup> UpsellType Values

Value	Gateway Constant Name	Description
0	UPSELL_TYPE_REPLACEMENT	The PLU was added as a replacement for the original PLU chosen.
1	UPSELL_TYPE_ADD_ON	The PLU was added to the transaction in addition to the original option chose.

<sup>2</sup> UpsellSalesChannelType Values

Value	Gateway Constant Name	Description
0	sctPOS	The upsell was performed from the Point of Sale
1	sctOrderEntry	The upsell was performed from Order Entry
2	sctKiosk	The upsell was performed from the Kiosk
3	sctWebStore	The upsell was performed from the WebStore

<sup>3</sup> UpsellStatus Values

Value	Gateway Constant Name	Description
0	UPSELL_STATUS_NO_UPSELL	Upsell was not offered for this order line
1	UPSELL_STATUS_UPSELL_SELECTED	Upsell was offered and selected for this order line
2	UPSELL_STATUS_UPSELL_CANCELLED	Upsell was offered, but not selected for this order line

## 15.64 OrderPayments

This table stores information on payments made by customers on an Order.

### Columns

Column	Type	Allow Nulls	Description
OrderPaymentID	Int	N	Primary key, always unique.
OrderLineID	Int	N	Foreign key to OrderLines.OrderLineID.
PaymentDate	DateTime	Y	The date the payment was made to the order.
PaymentFOP	Int	Y	The form of payment used to make the payment.
PaymentAmount	Money	Y	The amount of the payment.
CreditMemoID	Int	Y	If the form of payment is a credit memo, this contains the credit memo number. Otherwise, this is zero or NULL.
Endorsement	VarChar(50)	Y	Endorsement (i.e. card #, check account #) for this form of payment.
ExpirationDate	Char(4)	Y	Expiration date for credit/debit card, in the format MMYY.
OrderTransactionID	Int	Y	Foreign key to OrderTransactions.OrderTransactionID
Company	Int	Y	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
Category	Int	Y	The transaction account category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SubCat	Int	Y	The transaction account sub-category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated .
PayOnIssuance	Bit	N	If 1, the payment is a Pay On Issue payment  If 0, the payment is not a Pay On Issue payment (default)
CurrencyKey	Char(1)	Y	Key ('A'..'Z') representing currency used for payment. Any blank, space or null value represents base currency.
CardHolderName	VarChar(30)		Card holders name
AuthCode	VarChar(128)	Y	Approval Code (returned from host). <sup>1</sup>
TransID	Char(15)	Y	Transaction ID (returned from host). <sup>1</sup>
Validation	Char(4)	Y	Validation Code (returned from host). <sup>1</sup>
PSI	Char(1)	Y	Payment Service Indicator (returned from host). <sup>1</sup>
GxKeyID	Int	N	Foreign key to GxKeys.GxKeyID, defines encryption scheme used to encrypt values for this entry. This column will be zero if no columns are encrypted.
SearchEndorsementValue	VarChar(50)	Y	This field stores the last 4 characters of Endorsement. The value stored in this field is not encrypted, regardless of whether or not encryption is enabled for the associated form of payment.
CVNResultCode	Char(1)	Y	Result of CVN verification.
AVSResultCode	Char(5)	Y	Result of AVS verification.  <b>NOTE:</b> Galaxy POS does not update this column. It is populated by eGalaxy Server when importing an order from the eGalaxy Web Store.
HostTraceNumber	Char(8)	Y	Store BCAM response data.
StoredValueBalance	Float	Y	Used to store the balance of the stored value card at the time of the payment.
Bin	Char(6)	Y	Contains the BIN (the first 6 digits) of the card number used for the payment, if the "Collect Endorsement BINs" configuration option is turned on.
CardToken	NVarChar(62)	Y	Holds card token data returned from a credit card host system.

### Indexes

Name	Kind	Columns	Purpose
PKOrderPaymentsOrderPaymentID	P	OrderPaymentID	Primary Key.
IXOrderPaymentsOrderTransID		OrderTransactionID	Search payments by the OrderTransactionID
IXOrderPaymentsEndorsement		Endorsement	Kiosk Credit Card order search
IXOrderPaymentsLastUpdate		LastUpdate	Optional Index to improve performance of DB Update Process in DeleteCCInfoUtility.
IXOrderPaymentsOrderLineTranID		OrderLineID, OrderTransactionID	Index used in Contract Payments report.

<sup>1</sup> Note: Depending upon the credit card protocol being used, the format of these fields can vary. For certain protocols, some of these fields may always be empty or zero.

## 15.65 OrderRateTables

This table stores Sales Program rate tables. This is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
OrderRateTableID	Int	N	Primary key, always unique.
ItemRateID	Int	N	Foreign key to ItemRates.ItemRateID. This is the item rate record, of which this rate table is a part.
UpToScale	Int	N	The highest quantity to which this line in the rate table applies.
Kind	Int	N	Indicates how the price is calculated. <sup>1</sup>
Amount	Float	N	The actual percent or currency amount to apply to the price of the PLU, depending on the RateType. <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderRateTablesOrdRateTblID	P	OrderRateTableID	Primary Key.
IXOrderRateTablesItemRate	F	ItemRateID	Used to get the rate table records for a particular Item Rate.

<sup>1</sup> Kind Values

Value	Description	Amount
0	Flat price	The amount to charge the customer. Replaces the price of the ticket or retail item.
1	Currency amount discount	The amount to subtract from the price of the ticket or retail item.
2	Percentage discount	The percentage (12.34 = 12.34%) discount to apply to the price of the ticket or retail item.
3	Price Schedule	The amount to charge is determined by a price schedule. Replaces the price of the ticket or retail item.

## 15.66 OrderReprintHistory

This table stores order reprint information. This is used by Order Entry and Kiosk.

### Columns

Column	Type	Allow Nulls	Description
OrderReprintHistoryID	Int	N	Primary key, always unique.
OrderID	Int	Y	Foreign key to Orders.OrderID.
Node	Int	Y	Node that reprinted the order.
UserID	Int	Y	User that reprinted the order.
Source	Int	Y	Indicates how the order was reprinted. <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderReprintHistoryID	P	OrderReprintHistoryID	Primary Key.
IXOrderReprintHistoryOrderID	F	OrderID	Used to query table for reprints of an order.

<sup>1</sup> Source

Value	Description	Amount
0	Kiosk	The order was reprinted from the Kiosk. Use constant ORDER_HISTORY_FROM_KIOSK.
1	Order Entry	The order was reprinted from Order Entry. Use constant ORDER_HISTORY_FROM_OE.

## 15.67 OrderRules

This table is used to store a set of rules that are followed when creating new orders.

### Columns

Column	Type	Allow Nulls	Description
OrderRuleID	Int	N	Primary key, always unique.
Description	Varchar(40)	Y	Rule name description
CustomerContactAction	Int	Y	Action for Customer Contact <sup>1</sup>
BillToContactAction	Int	Y	Action for Bill-To Contact <sup>1</sup>
ShipToContactAction	Int	Y	Action for Ship-To Contact <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKOrderRulesOrderRuleID	P	OrderRuleID	Primary Key.

<sup>1</sup> CustomerContactAction, BillToContactAction, and ShipToContactAction Values

Value	Gateway Constant Name	Description
1	ORDER_RULE_ALWAYS_NEW	Always create a new contact
2	ORDER_RULE_USE_PRIMARY	Use the defined primary contact
3	ORDER_RULE_SELECT_LIST	Select the contact from a list
4	ORDER_RULE_CUSTOMER	Set the contact to be the same as the customer contact

## 15.68 Orders

This table stores general information about each Order.

### Columns

Column	Type	Allow Nulls	Description
OrderID	Int	N	Primary key, always unique.
CustomerID	Int	N	Foreign key to Customers.CustomerID, representing the customer who created the Order. Must contain a value greater than zero, and point to a valid customer.
AccountID	Int	Y	Foreign key to ARAccounts.AccountID, representing the Charge Account responsible for paying for the Order. This field is zero (0) or NULL if the customer does not have charge account privileges.
ContactID	Int	N	Foreign key to CustContacts.CustContactID, representing the person to write, telephone, or fax concerning the Order. Must contain a value greater than zero, and point to a valid customer contact.
BillToContactID	Int	Y	Foreign key to CustContacts.CustContactID, representing the person to whom the invoice or credit memo should be sent. This field is zero (0) or NULL if the customer does not have charge account privileges.
ShipToContactID	Int	Y	Foreign key to CustContacts.CustContactID, representing the person to whom the Order will be sent.
GroupVisitID	Int	Y	Foreign key to GroupVisits.GroupVisitID, representing the arrival date and number of people arriving on this order. This field is zero (0) or NULL if the order has no arrival information.
OrderStatus <sup>1</sup>	Int	N	The current status of the order.
OpenDate	DateTime	N	The date the order was created.
NoteID	Int	Y	Foreign key to Notes.NoteID.
Reference	Char(40)	Y	Additional information about the order.
PO	Char(20)	Y	The customer's purchase order number for the order.
UnissuedQty	Int	Y	The number of tickets (or retail items) not yet given to the customer.
UnissuedAmt	Money	Y	The currency amount of tickets (or retail items) not yet given to the customer.
Balance	Money	Y	The currently-due currency amount of the invoice. This amount changes as payments are made.
BatchPrintDate	DateTime	Y	The date the order was last Batch Printed.
DeliveryMethod	Int	Y	FK reference to DeliveryMethods.DeliveryMethodID  (obsolete field, preserved for backward compatibility)
DeliveryMethodID	Int	Y	FK reference to DeliveryMethods.DeliveryMethodID
DeliveryDetails	VARCHAR(255)	Y	Specific delivery details, used particularly when the Delivery Method is "Other".
SPConnectionID	Int	Y	SPConnectionID is currently being kept for historical reasons but should no longer be used to look up the Sales Program information, use SalesProgramID instead.
TrackingNbr	Char(30)	Y	Additional Delivery information.
PickupDate	DateTime	Y	Date and time the order was picked up by the customer.
ExternalID	NVarChar(40)	Y	Additional order information relating to an external system.
UnformattedExternalID	NVarChar(40)	Y	A copy of the ExternalID without any formatting.
OrderAmt	Money	Y	The original currency amount of the order. This is different from the Balance, in that it does not change as payments are made to the order.
PickupNode	Int	Y	Node where the order items were picked-up
PickupUserID	Int	Y	UserID that was logged at the pickup time and gave the tickets to the customer.
Secure	Bit	Y	Is an order marked as secure or not. If set to yes (true), only the authorized users can modify the Order.
PayOnIssuance	Bit	Y	If 1, order entry applies payment when the order is printed. Order is marked as Pay On Issue  If 0, order entry does not apply payment when the order is printed. Order is not marked as Pay On Issue
OrderResolution <sup>3</sup>	Integer	Y	Specifies how the order was closed
TaxStatus <sup>4</sup>	Integer	Y	Tax status for the order. If this is set to non-taxable, the Order is non-taxable regardless of any other system settings (item, customer, etc).
TotalDiscount	Money	Y	Total Discount amount for the Order
TotalPayment	Money	Y	Total Payment amount for the Order
TotalTax	Money	Y	Total Tax amount for the Order
SalesCategoryGroupID	Int	Y	FK reference to SalesChannelDetails.SalesChannelDetailID
PromotionID	Int	Y	FK reference to the Promotions.PromotionID table  ID of the promotion used by the guest for the purchase (on the web store)
PromotionCode	VARCHAR(50)	Y	Promotion code used by the guest for the purchase (on the web store)
MinimumPaymentDue	Money	Y	Payment amount due for an order that has tickets/passes with Payment Contracts
AttributeValueGroupID	Int	Y	Points to the Attribute Values Group
SalesProgramID	Int	Y	Foreign Key to Sales Programs.SalesProgramID
CustCategoryID	Int	Y	FK reference to CustCategories.CustCategoryID, to be used in Associated Tickets processing
TotalPromotionalDiscount	Money	Y	Total amount of promotional discount applied to the order
PermitReturnsOnly	Bit	Y	This flag locks the order down; preventing any functional modifications to the order, except returns. Returns must be made to the FOP that was used to purchase the order.
TotalRedeemedValue	Float	Y	Total points redeemed for the order
LastMenuID	Char(16)	Y	Menu ID of the POS menu that was selected when the last auto-issue item was selected in the order.
GroupSalesCode	VARCHAR(30)	Y	The groups sales code that was entered when placing a web order.
ViewOrderWebURL	VARCHAR(500)	Y	URL to the webstore View Order page, so the user has a link to review their order or reprint tickets in the order.
LoyaltyAccountNo	VARCHAR(40)	Y	The Loyalty account number currently in use in the order.
LoyaltyProgramID	Integer	Y	The Loyalty Program ID currently in use in the order.
PendingLoyaltyBonusPoints	Float	Y	Stores the pending bonus points on an order.
IssuedLoyaltyBonusPoints	Float	Y	Stores the issued bonus points on an order.
Locked	Bit	Y	Flag indicates if the order has been locked. No accounting transactions can be performed on an order when locked. The ability to lock or unlock

			is restricted by user privileges.
GiftAidStatus <sup>5</sup>	Integer	Y	Indicates whether or not the guest has been asked to participate in Gift Aid.
NodeNo	Int	Y	Node that created the order
Salesperson	NVarChar(50)	Y	Sales person that created the order
TranslationLanguageID	Int	Y	FK to TranslationLanguages.TranslationLanguageID
RemoteAddress	VarChar(500)	Y	The IP address of the client associated with the Order
PersonalMessageID	Integer	Y	Foreign key to Notes.NoteID
SecureToken	VarChar(12)	Y	An encrypted string used to identify the order alongside the order ExternalID and UniqueID under certain configuration conditions.
OrderGUID	uniqueidentifier	Y	GUID associated to this order. Used to uniquely identify this order across any system.
OrderHash	nvarchar(500)	Y	String hash value generated from various order data components. Used to uniquely identify this order across any system.
LockedPaymentState <sup>7</sup>	Integer	Y	If the order is locked, this contains the payment state of the order at the time it was locked. This field is zero (0) or NULL if the order is not locked.

The following columns are used only by Web Store.

TotalShipping | Money | Y | Total shipping charge |  
 MerchantID | Int | Y | MerchantID - Foreign Key to Merchants Table |  
 PromotionOfferID | Int | Y | PromotionOfferID - foreign key to PromotionOffers table |  
 SessionID | Int | Y | ID from eGalaxy for Capacity Managed Tickets |  
 ExternalOrderStatus <sup>6</sup> | Int | Y | eGalaxy PickupOrder status and Web Store order status |  
 ASPNetSessionID | Varchar(200) | Y | The session ID associated with the Order. |  
 ServerAddress | Varchar(500) | Y | The address of the server processing the ASP.NET session. |  
 Workflow | NVarChar(max) | Y | Stores the workflow object that is used to re-inflate a session coming back from a 3rd party payment provider. |  
 eGalaxySource | NVarChar(128) | Y | The external eGalaxy Source ID that was used for the eGalaxy Messaging in this order. This is the eGalaxy Source ID configured in the Web Admin Panel unless there was an eGalaxy Source configured on the Category Group in Sales Channel Manager. |

## Indexes

Name	Kind	Columns	Purpose
PKOrdersOrderID	P	OrderID	Primary Key.
IXOrdersCustomerID	F	CustomerID	Used to select a list of orders by customer name or account.
IXOrdersAccount	F	AccountID	Used to select a list of orders by account.
IXOrdersStatus		OrderStatus	Used to get a list of all open or closed orders.
IXOrdersOpenDate		OpenDate	Used to select a list of orders that fall within a particular range of open dates.
IXOrdersBatchPrintDate		BatchPrintDate	Used to select orders for the Batch Print Report and the Order Breakage Report.
IXOrdersOrderIDRecVer		OrderID, RecordVersion	Used to speed-up Auto Batch Print.
IXOrdersPickupDateNodeUserID		PickupDate PickupNode PickupUserID	Used via the TOrdersList Business Object to generate the Order Pickup Report that lists the Orders that have just been picked-up (in this case the filter includes all the 3 fields) OR the orders picked-up in a specified date/time range.
IXOrdersExternalID		ExternalID	Used when searching orders using the Quick Pickup Galaxy function
IXOrdersArrivalSummary		GroupVisitID	Arrival Summary Report
IXOrdersContactDelMethStatus		ContactID, OrderID, DeliveryMethodID, OrderStatus, UnissuedQty, Balance	To improve query performance when searching for orders by contact.
IXOrdersDeliveryMethodID		DeliveryMethodID	To improve query performance for kiosk order pickup
FIOrdersKioskPickup		OrderStatus, UnissuedQty, Balance	Filtered index to improve query performance for kiosk order pickup, using clause WHERE OrderStatus < 3 and UnissuedQty > 0 and Balance <= 0;
IXOrdersBillToContactID		BillToContactID	To improve performance of queries that search for orders on BillToContactID
IXOrdersShipToContactID		ShipToContactID	To improve performance of queries that search for orders on ShipToContactID
<b>Web Store Specific Indexes</b>			
IXOrdersExternalOrderStatus		ExternalOrderStatus	Used for eGalaxy to WebStore Order Pickup function
IXOrdersSessionIDExternalOrderStatus		SessionID, ExternalOrderStatus	Used by WebStore:SELECT Count(*) as OrderCount FROM Orders WHERE SessionID=95405 AND (ExternalOrderStatus=1 OR ExternalOrderStatus=3)
IXOrdersUnformattedExternalID		UnformattedExternalID	Search by UnformattedExternalID in Kiosk

## <sup>1</sup> OrderStatus Values

Value	Gateway Constant Name	Description
1	ID_QUOTE_ORDERSTATUS	Quote.
2	ID_OPEN_ORDERSTATUS	Open.
3	ID_CLOSED_ORDERSTATUS	Closed.
4	ID_INVOICED_ORDERSTATUS	Invoiced.
5	ID_EGX_IMPORT_PLACEHOLDER_ORDERSTATUS	Placeholder for eGX orders.

## <sup>2</sup> DeliveryMethod Values.

Value	Gateway Constant Name	Description
0	DM_NONE	None
1	DM_HOLD	Hold.
2	DM_MAIL	Mail.
3	DM_OVERNIGHT	Overnight.
4	DM_SECONDDAY	Second Day.
5	DM_OTHER	Other.

## <sup>3</sup> OrderResolution Values

--

Value	Gateway Constant Name	Description
0	ORD_RES_NONE	None
1	ORD_RES_ORDERENTRY	Order was closed by user in OE
2	ORD_RES_QUICKPICKUP	Order was closed by Quick Order Pickup function
3	ORD_RES_eGALAXY	Order was closed by eGalaxy
4	ORD_RES_OVERDUE	Order was closed because it was overdue for pickup
5	ORD_RES_INVOICED	Order was closed by invoicing

**4 TaxStatus Values**

Value	Gateway Constant Name	Description
0	TAX_STATUS_ENABLED	Order is taxable
1	TAX_STATUS_DISABLED	Order is non-taxable

**5 GiftAidStatus Values**

Value	Gateway Constant Name	Description
0	gasNotAsked	Guest has not been asked to participate in Gift Aid.
1	gasAccepted	Guest has agreed to participate Gift Aid.
2	gasDeclined	Guest has declined to participate in Gift Aid.
3	gasNonGiftAid	Guest has agreed to make a non-Gift Aid donation.

Web Store specific

**6 ExternalOrderStatus Values**

Value	Gateway Constant Name	Description
0	Unprocessed	Order has not been completed (sometimes referred to as a "cart" or incomplete order)
1	New	New order is ready for eGalaxy Server pickup
2	PickedUp	The order has been picked-up by eGalaxy Server
3	Error	There was error while processing the order by eGalaxy Server
4	ResellerProcessed	This order is processed by Reseller Web store application
5	PickupInProgress	This order is in the process of pickup. A pickup order request is in process of collecting data for this order
6	OrderSoftDeleted	This order has been deleted and will not show up in Galaxy any longer, but remains in the database for auditing purposes.
40	ProcessingOrder	Temporary indicator to prevent order from being processed twice
41	AuthInProgress	Temporary indicator (should get set to 'AuthApproved' next)
42	AuthApproved	Temporary indicator (should get set to 'New' next)
43	AuthDeclined	Temporary indicator that auth was declined. Allow customer to try again.
44	AuthError	Temporary indicator used to determine if error message is displayed. Customer is not allowed to try again.
45	PaymentFraudPreventionFailed	Temporary indicator that payment fraud was rejected. Allow customer to try again.
50	ApprovallInProgress	Temporary indicator used by indirect confirm providers, i.e. PayPal
51	ApprovalComplete	Temporary indicator used by indirect confirm providers, i.e. PayPal
52	ReapprovallInProgress	Temporary indicator used by indirect confirm providers, i.e. PayPal
100	AccertifyReadyForPickup	
101	AccertifyPickupInProgress	
102	AccertifyProcessingError	
110	AccertifyRecommendedApprove	
111	AccertifyRecommendedReject	
112	AccertifyRecommendedReview	
113	AccertifyRejectVoidPending	
114	AccertifyRejectNeedsManualVoid	
1000	OrderDumpedDueToError	Used when the order has to be dumped due to an error, most typically a CartTrans error

**7 LockedPaymentState Values**

Value	Gateway Constant Name	Description
0	Unknown	The payment state of order has not been determined. Orders that are not locked will use either this value or NULL.
1	UnsupportedAccount	The account on the order does not support the retrieval of the payment state.
2	Paid	The order is completely paid.
3	UnpaidCredit	The order is not completely paid, and the account has credit available.
4	UnpaidDue	The order is not completely paid, and it either has no account available or the account balance has surpassed the credit limit.

## 15.69 OrderSurveys

The eGalaxy webstore can capture a consumer opt-in to participate in a Guest Visit Survey. The OrderSurveys table is used to track the state of the survey in its process flow from opt-in of the survey, through sending of a link to take the survey, to completion of the survey.

### Columns

Column	Type	Allow Nulls	Description
OrderSurveyID	Int	N	Primary key, always unique
OrderID	Int	N	FK reference to Orders table
SurveyDefinitionID	Int	N	FK reference to Surveys table
ContactID	Int	N	FK reference to CustContacts table
SurveyStatus	Int	N	The status or state of the Survey <sup>1</sup>
SurveyResultID	Int	N	FK reference to the SurveyResults table
RequestSurveySendDate	DateTime	Y	The date and time that PostUsageService calculated the survey can be sent out.
SurveyCompletionDate	DateTime	Y	Date the Survey was received by the contact
ResponseSurveySendDate	DateTime	Y	The date and time that the SurveyScorerService sent out a follow up email based on the score of the survey.
JnlTranID	Int	N	FK reference to the JnlSurvey.JnlTranID to allow the answers to be scored later.
SurveyScore	Float	Y	The positive score of the results of the survey.
RequestSurveyMessageQueueID	Int	Y	FK to MessageQueue.MessageQueueID for the email sent from PostUsageService.
ResponseSurveyMessageQueueID	Int	Y	FK to MessageQueue.MessageQueueID for the email sent from SurveyScorerService.
SurveyURL	Varchar(255)	Y	The URL where the customer can take the web survey.

### Indexes

Name	Kind	Columns	Purpose
PKOrderSurveyID	P	OrderSurveyID	Primary Key

### <sup>1</sup> SurveyStatus Values

Value	Gateway Constant Name	Description
0	ossNew	A new order has been created that contained a guest's opt-in to the Guest Visit Survey.
1	ossSent	The email with the link to take the survey has been sent to the Contact's email address.
2	ossReceived	The survey answers have been received from the eGalaxy Survey Processor.
3	ossScored	The survey has been successfully scored by the Survey Grading Service
999	ossError	An error occurred while attempting to process the Order Survey.

## 15.70 OrderTaxDetails

Contains the information for taxes journalized during the issuing of an order's tickets and/or retail items.

### Columns

Column	Type	Allow Nulls	Description
OrderTaxDetailID	Int	N	Primary key, always unique.
OrderLineID	Int	Y	Foreign key to OrderLines.OrderLineID. The unique id of the order line to which this detail refers.
TaxID	Int	Y	The unique id of the tax (currently 1 thru 8)
Amount	Money	Y	The total amount of the tax
OrderTransactionID	Int	Y	Foreign key to OrderTransactions.OrderTransactionID
Company	Int	Y	The ID number of the company receiving revenue for the issued ticket, as specified in the Chart of Accounts entry to which this ticket is associated.
Category	Int	Y	The transaction account category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.
SubCat	Int	Y	The transaction account sub-category for the Company, as specified in the Chart of Accounts entry to which this ticket is associated.

### Indexes

Name	Kind	Columns	Purpose
PKOrderTaxDetailsOrderTaxDtlID	P	OrderTaxDetailID	Primary Key.
IXOrderTaxDetailsOrderLineID		OrderLineID	Search taxes by OrderLineID
IXOrderTaxDetailsOrderTransID		OrderTransactionID	Search taxes by OrderTransactionID

## 15.71 OrderTransactions

Associates an Order with a Journal Transaction. An order can have multiple transactions, but each transaction can have only one order.

### Columns

Column	Type	Allow Nulls	Description
OrderTransactionID	Int	N	Primary key, always unique.
OrderID	Int	Y	Foreign key to Orders.OrderID
NodeNo	Int	Y	The number of the node that generated this transaction
TransactionNumber	Int	Y	Number of the transaction generated by this order
TransactionDate	datetime	Y	Date and time of this transaction
UserID	int	Y	ID of the user who processed this transaction
Status	Int	Y	The status of this transaction. <sup>1</sup>
VoidedTransactionID	Int	Y	Foreign key to OrderTransactions.OrderTransactionID. This points to the transaction which this transaction is voiding.
ShiftNo	Int	Y	The shift number for this transaction.
BatchID	Int	Y	If greater than zero, this is a batch print transaction

### Indexes

Name	Kind	Columns	Purpose
PKOrderTransOrderTransactionID	P	OrderTransactionID	Primary Key.
IXOrderTransOrderIDTransNum	IX	OrderID & TransactionNumber	Search for a transaction by order id and journal transaction number. Used to fill-in the VoidedTransactionID for a void.
IXOrderTransTransDate	IX	TransactionDate	Index for BatchPrint report
IXOrderTransTransNumNodeNo	IX	TransactionNumber, NodeNo	For Journal Sender plug in

<sup>1</sup> Status Values

Value	Gateway Constant Name	Description
0	OT_NONE	This transaction has not been voided, and is not voiding another transaction.
1	OT_VOID	This transaction is voiding another one.
2	OT_VOIDED	This transaction has been voided.
3	OT_VOID_ERROR	An error occurred during an attempt to void this transaction. It may have been incompletely voided.

## 15.72 PhoneNumbers

When system configured to use option for multiple phone numbers, this table stores various phone numbers for contacts and customers.

### Columns

Column	Type	Allow Nulls	Description
PhoneNumberID	Integer	N	Primary Key.
PhoneType	Integer	Y	Indicates the type of phone number. <sup>1</sup>
Phone	Char(30)	Y	The phone number.
UnformattedPhone	VARCHAR(30)	Y	The phone number without any formatting.
AllowContact	Bit	Y	Supports the ability to save Contact's preferences regarding communication methods. Indicates whether or not the contact may be contacted via this phone number.
ContactID	Integer	Y	Link to CustContacts.CustContactID, each record has either a ContactID or a Customer ID
CustomerID	Integer	Y	Link to Customers.CustomerID, each record has either a ContactID or a Customer ID

### Indexes

Name	Kind	Columns	Purpose
PKPhoneNumberID	P	PhoneNumberID	Primary Key.
IXPhoneNumbersContactID	IX	ContactID	Supports searching by ContactID
IXPhoneNumbersCustomerID	IX	CustomerID	Supports searching by CustomerID
IXUnformattedNumber	IX	UnformattedNumber	Improve query performance

<sup>1</sup> PhoneType Values

Value	Gateway Constant Name	Description
0	ptHome	Home phone number
1	ptWork	Work phone number
2	ptMobile	Mobile phone number
3	ptFax	Fax phone number
4	ptOther	Other phone number

### 15.73 SalesPrograms

A Sales Program consists of zero or more item constraints, and/or zero or more discount rates, that are effective over a specific period of time. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

#### Columns

Column	Type	Allow Nulls	Description
SalesProgramID	Int	N	Primary key, always unique.
Name	Char(20)	N	Name of the Sales Program.
Description	Char(50)	N	Description of the Sales Program.
BeginDate	DateTime	Y	Date the sales program begins.
EndDate	DateTime	Y	Date the sales program ends.
ValidOnSundays	Bit	N	If TRUE, indicates that the sales program should apply on Sundays. <i>This column is not currently used by the system.</i>
ValidOnMondays	Bit	N	If TRUE, indicates that the sales program should apply on Mondays. <i>This column is not currently used by the system.</i>
ValidOnTuesdays	Bit	N	If TRUE, indicates that the sales program should apply on Tuesdays. <i>This column is not currently used by the system.</i>
ValidOnWednesdays	Bit	N	If TRUE, indicates that the sales program should apply on Wednesdays. <i>This column is not currently used by the system.</i>
ValidOnThursdays	Bit	N	If TRUE, indicates that the sales program should apply on Thursdays. <i>This column is not currently used by the system.</i>
ValidOnFridays	Bit	N	If TRUE, indicates that the sales program should apply on Fridays. <i>This column is not currently used by the system.</i>
ValidOnSaturdays	Bit	N	If TRUE, indicates that the sales program should apply on Saturdays. <i>This column is not currently used by the system.</i>
ValidOnHolidays	Bit	N	If TRUE, indicates that the sales program should apply on holidays, regardless of the day of week the holiday falls on. <i>This column is not currently used by the system.</i>
RestrictItems	Bit	N	If TRUE, indicates that the items available for sale are restricted to those in the Item Constraints. If FALSE, it excludes those in the Item Constraints
SalesProgramTypeID	Int	Y	Foreign key to SalesProgramTypes.SalesProgramTypeID.
RestrictWillCallOrders	Bit	Y	When set to TRUE (1), restricts user to save an order if the delivery method of the order is set to "hold".
Consignment	Bit	Y	If TRUE (1), all the tickets sold under this Sales Program are considered as Consignment Tickets
LegacyTableRateInPOS	Bit	Y	If TRUE (1), Point of Sale will revert to legacy (pre-7.3.0.25) behavior when handling Sales Programs with table rates.
SalesProgramGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSalesProgramsSalesProgramID	P	SalesProgramID	Primary Key.
IXSalesProgramsIDDatesItems		SalesProgramID, BeginDate, EndDate, RestrictItems	Reduce query time

## 15.74 SalesProgramTypes

This table stores the Sales Program types. A Sales Program Type is a way of categorizing the Sales Programs. Each Sales Program Type will consist of a name and a description and a unique id.

### Columns

Column	Type	Allow Nulls	Description
SalesProgramTypeID	Int	N	Primary key, always unique.
Name	Varchar(20)	Y	Name of the type
Description	Varchar(40)	Y	Description of the type

### Indexes

Name	Kind	Columns	Purpose
PKSalesProgTypesSalesProgTypeID	P	SalesProgramTypeID	Primary Key.
IXSalesProgramTypesName		Name	User-friendly sort by name.

## 15.75 SerializedStatements

This table stores information about the serialized numbers and serialized statements for invoices, and credit memos. The SerialNumber will be a unique sequential number for each Year/CompanyID/MediaID combination.

### Columns

Column	Type	Allow Nulls	Description
SerializedStatementID	Int	N	Primary key, always unique.
Year	Int	N	Year of the serialized value.
MediaID	Int	N	Foreign key to Media.MediaID.
SerialNumber	Int	N	The most recently used serial number for the given Company and MediaID
FullSerialNumber	Varchar(20)	N	The full serial number that was generated for the statement.
StatementDateTime	DateTime	N	The Date/Time that the statement was generated.
HTML	Varchar(MAX)	Y	The contents of the generated statement in HTML format.
CustomerID	Int	N	Link to Customers.CustomerID.
OrderID	Int	N	Link to Orders.OrderID. Zero will be used if no order is associated.
ReferenceTableID	Int	N	A specific table that the statement is associated to. <sup>1</sup> The row in this table is specified by ReferenceID.
ReferenceID	Int	N	The specific row ID that the statement is associated to. For example, if ReferenceTableID = 108 then the statement references the Orders table where OrderID = ReferenceID.
CompanyID	Int	N	Foreign key to Companies.CompanyID. Zero indicates that the configuration is set to not include Company in Serialized Statement Serial Numbers.
DepositAmount	Money	Y	Stores the amount owed for this statement when the statement was generated.
PaymentDate	DateTime	Y	Stores the due date of the payment for this statement.
DepositPercentage	Float	Y	Stores the down payment percentage amount selected by the user for this statement.
RemainingBalance	Money	Y	Stores the remaining balance after down payment amount for this statement.
StatementType	Int	Y	Represents the type of the serialized statement. <sup>2</sup>
Status	Int	Y	The status of this statement. <sup>3</sup>
CancelDateTime	DateTime	Y	The date/time this statement was canceled, if canceled.
CancelsSerializedStatementID	Int	Y	If this statement is a cancellation of another statement, the SerializedStatementID of the statement this statement cancels.
OrderTotal	Money	Y	Stores a snapshot of the order total at the time this serialized statement was generated.

### Indexes

Name	Kind	Columns	Purpose
PKSerializedStatementsID	P	SerializedNumberID	Primary Key.
IXYearCompanyIDMediaIDSerialNumber		Year, CompanyID, MediaID, SerialNumber	Reduce query/lock time when serializing statements.

<sup>1</sup> Table ID Values

TableID	Table
108	ORDERS
122	ARINVOICES
124	ARCREDITMEMOS

<sup>2</sup> StatementType Values

Value	Gateway Constant Name	Description
0	sstUnspecified	The statement type was not specified
1	sstDeposit	This is a deposit statement
2	sstFinal	This is a final statement
3	sstCancellation	This is a cancellation statement

<sup>3</sup> Status Values

Value	Gateway Constant Name	Description
0	sssValid	Valid
1	sssCancel	Canceled

## 15.76 SerializedStatementDetails

The SerializedStatementDetails table contains a snapshot record for each ticket/item/fee/donation/total tax record in the order, taken at the time the statement was generated, and based on the deposit amount/percentage for the serialized statement.

### Columns

Column	Type	Allow Nulls	Description
SerializedStatementDetailID	Int	N	Primary key, always unique.
SerializedStatementID	Int	N	FK to the SerializedStatement record for this detail
DetailType	Int	N	The type of this detail. <sup>1</sup>
Amount	Money	N	The amount for this detail for this serialized statement. For example, if the serialized statement is a 50% statement, this amount is 50% of this ticket/item/tax.
OrderLineID	Int	Y	The order line this detail is associated with.
TaxID	Int	Y	If this detail record is of type 9 (total tax), which tax # this tax represents (1-8).
AccountID	NVarChar(12)	Y	If this detail record is a ticket/item/fee/donation, the AccountID for the ticket/item/fee/donation.

### Indexes

Name	Kind	Columns	Purpose
PKSerializedStatementDetailID	P	SerializedStatementDetailID	Primary Key

<sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
1	ssdtItem	Item Detail
8	ssdtPass	Pass Detail
9	ssdtTotalTax	Total Tax Detail
10	ssdtPackage	Package Detail
11	ssdtPackageDetail	Package Detail Detail
12	ssdtFee	Fee Detail
14	ssdtDonation	Donation Detail

## 15.77 SMSTemplates

The SMSTemplates table contains templates for SMS messages that will be sent by Galaxy for either order confirmation or fulfillment messages.

### Columns

Column	Type	Allow Nulls	Description
SMSTemplateID	Integer	N	Primary key. Obtained from GatewayCounters.
Name	NVarChar(200)	N	Name of this SMS template.
Provider	Int	N	Provider to use for this SMS template. <sup>1</sup>
Text	NVarChar(max)	N	Content of this SMS message.

### Indexes

Name	Kind	Columns	Purpose
PKSMSTemplateID	P	SMSTemplateID	Primary Key

<sup>1</sup> Provider Values

Value	Gateway Constant Name	Description
0		None
1		Hengrui

## 15.78 SPConnections

This table relates Sales Programs to Customers or to Customer Categories. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
SPConnectionID	Int	N	Primary key, always unique.
SalesProgramID	Int	N	Foreign key to SalesPrograms.SalesProgramID.
ConnectionType	Int	N	Type of record associated with this sales program. <sup>1</sup>
ConnectionID	Int	N	Foreign key to Customers.CustomerID or CustCategories.CustCategoryID, depending upon the value in the ConnectionType column.
Priority	Integer	Y	Priority of the SPConnection for a Customer or Customer Category
SPConnectionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKSPConnectionsSPConnectionID	P	SPConnectionID	Primary Key.
IXSPConnectionsSalesProgramID	F	SalesProgramID	Used to select sales program information for a particular connection record.
IXSPConnectionsConnectionID	F	ConnectionID	Used to select customer or customer category information for a particular connection record.
IXSPConnectionsTypeIDSPID		ConnectionType, ConnectionID, SPConnectionID	Reduce query time

<sup>1</sup> ConnectionType Values

Value	Description
1	ConnectionID is a foreign key reference to Customers.CustomerID.
2	ConnectionID is a foreign key reference to CustCategories.CustCategoryID.

**15.79 SPPriceCalendars**

Connects a price program to a sales program.

**Columns**

Column	Type	Allow Nulls	Description
SPPriceCalendarID	Int	N	Primary key, always unique
SalesProgramID	Int	N	FK reference to SalesPrograms.SalesProgramID column
PriceProgramID	Int	N	FK reference to PricePrograms.PriceProgramID column
EffectiveDateTime	DateTime	N	Effective Date/Time of the PriceProgram
PriceProgramGroupID	Integer	N	Foreign key to PriceProgramGroups.PriceProgramGroupID. Indicates the sales program price calendar is associated with a price program group. Zero indicates no price program group is associated.
SPPriceCalendarGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**Indexes**

Name	Kind	Columns	Purpose
PKSPPriceCalendarID	P	SPPriceCalendarID	Primary Key.
IXSPPPriceProgEffectiveDateTime	U	SalesProgramID, PriceProgramID, EffectiveDateTime	Unique key
IXSPPPriceCalEffectiveDateTime		EffectiveDateTime	Used by the query to get Price program for a given date and time

**15.80 SPPriceSchedules**

Connects a sales program to a price program and time range.

**Columns**

Column	Type	Allow Nulls	Description
SPPriceScheduleID	Int	N	Primary key, always unique
ItemRateID	Int	N	FK reference to ItemRates.ItemRateID column
PriceProgramID	Int	N	FK reference to PricePrograms.PriceProgramID
PriceProgramTimeRangeID	Int	N	FK reference to PriceProgramTimeRanges.PriceProgramTimeRangeID column
Price	Money	N	Price to use for the given PLU
DisbursementID	Int	Y	FK reference to Disbursements.DisbursementID column
SPPriceScheduleGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

**Indexes**

Name	Kind	Columns	Purpose
PKSPPriceScheduleID	P	SPPriceScheduleID	Primary Key.
AKSPPriceSchedUniqueKey	U	ItemRateID, PriceProgramID, PriceProgramTimeRangeID	Unique key.

## 15.81 Statements

This table stores the file locations of order, invoice, and credit memo printing templates. It is used by Order Entry, and Galaxy POS (for printing invoices) when configured to use SQL **Customers**.

### Columns

Column	Type	Allow Nulls	Description
StatementID	Int	N	Primary key, Always unique
Description	Char(40)	N	Description
FilePath	VarChar(255)	N	Complete path and filename of the statement file
StatementDataType	Int	Y	<sup>1</sup> The type of statement that will be generated when using this record. This can be a statement generated by using a .DOT file or one generated by parsing a WebTemplate.
AuxID	Int	Y	FK to an auxiliary table that contains a template file to parse and generate the statement. Starting in 6.4, the AuxID value will be 0 when StatementType = 0 or > 0 when StatementType = 1 (FK to WebTemplates table).
Serialize	Bit	Y	Indicates that this statement should be serialized when applicable.
PromptForDepositAmount	Bit	Y	Set to 1 when Galaxy should prompt the user for a deposit amount when generating this statement.
PromptForPaymentDate	Bit	Y	Set to 1 when Galaxy should prompt the user for a payment date when generating this statement.

### Indexes

Name	Kind	Columns	Purpose
PKStatementsStatementID	P	StatementID	Primary Key

### <sup>1</sup> StatementDataType Values

Value	Gateway Constant Name	Description
0	OE_STATEMENT_DOC_TEMPLATE_TYPE	Statement is a Document Template
1	OE_STATEMENT_WEB_TEMPLATE_TYPE	Statement is a Web Template

## 15.82 TaxSetDetails

More than one tax can be applied to the tickets and retail items sold to a customer. These taxes are collected into "sets", which can then be associated with a customer record. This table contains the individual taxes that make up a Tax Set. It is used by Order Entry, and Galaxy POS when configured to use SQL [Customers](#).

### Columns

Column	Type	Allow Nulls	Description
TaxSetDetailID	Int	N	Primary key, Always unique.
TaxSetID	Int	N	Foreign Key to TaxSets.TaxSetID.
TaxID	Int	N	Reference to local Tax flat file corresponding to tax 1 through 8.
TaxSetDetailGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKTaxSetDetailsTaxSetDetailID	P	TaxSetDetailID	Primary Key.
IXTaxSetDetailsTaxSetID	F	TaxSetID	Used to select the tax set details for a particular tax set.

### 15.83 TaxSets

More than one tax can be applied to the tickets and retail items sold to a customer. These taxes are collected into "sets", which can then be associated with a customer record. This table stores descriptions for Tax Sets. It is used by Order Entry, and Galaxy POS when configured to use SQL **Customers**. The contents of a Tax Set can be found in the **TaxSetDetails** table.

#### Columns

Column	Type	Allow Nulls	Description
TaxSetID	Int	N	Primary key, Always unique.
Description	Char(20)	N	Tax Set description.
TaxSetGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKTaxSetsTaxSetID	P	TaxSetID	Primary Key.

## 15.84 TempSpendingHistory

This table temporarily stores journal header data when collecting the spending history for a contact or customer. It is utilized by the GTS\_SP\_GetSpendingHistory stored procedure.

### Columns

Column	Type	Allow Nulls	Description
SessionID	Char(50)	Y	Unique ID generated for data being retrieved.
JnlTranID	Int	Y	JnlHeader.JnlTranID
FiscalDate	Datetime	Y	JnlHeader.FiscalDate
TranDate	DateTime	Y	JnlHeader.TranDate
NodeNo	Int	Y	JnlHeader.NodeNo
TranNo	Int	Y	JnlHeader.TranNo
AgencyID	Int	Y	JnlHeader.Agency
FOPID	Int	Y	JnlPayments.FOPID
FOPName	Varchar(50)	Y	FOPs.FOPName
AgencyName	Varchar(20)	Y	Agencies.AgencyName

### Indexes

Name	Kind	Columns	Purpose
IXTSHSessionID	AK	SessionID	To facilitate finding data by SessionID
IXTSHJnlTranID	AK	JnlTranID	To facilitate finding data by JnlTranID

## 15.85 UserFieldLabels

This table contains prompts and data-entry max lengths for the User-Defined fields in the **Customers** table.

### Columns

Column	Type	Allow Nulls	Description
UDFLabelID	Int	N	Primary key, Always unique.
Label	Char(30)	N	This is the prompt presented to the user when editing or viewing the User Defined Field.
Length	Int	N	This is the maximum length of the data you can enter into the User Defined Field.
Required	Bit	Y	Flag indicating if this user defined field is required to save

### Indexes

Name	Kind	Columns	Purpose
PKUserFieldLabelsUDFLabelID	P	UDFLabelID	Primary Key.

## 16 Passes

The features of the Galaxy Photo Pass module provide extensive pass usage and valuable information about the activities performed by a pass holder. Galaxy Photo Pass allows the facility back office to utilize important pass information for reporting purposes.

## 16.1 JointMembers

The JointMembers table contains information about joint members associated with a pass record.

### Columns

Name	Type	Allow Nulls	Description
JointMemberID	Integer	N	Primary key, always unique
PassNo	Integer	N	Pass the joint member is associated to. Foreign key to Passes.PassNo.
ContactID	Integer	N	Contact containing demographic information about this joint member. Foreign key to CustContacts.CustContactID.
VisualID	VarChar(40)	Y	Visual ID for the membership card associated with this member.
PictureID	Integer	Y	Picture associated with this member. Foreign key to Pictures.PictureID.
PreviousID	Integer	Y	The JointMemberID of the previous joint member record in the chain. When a joint membership is renewed, reissued/reprinted, or upgraded, this points to the previous joint member record for the same member.
MemberType	Integer	N	<sup>1</sup> Indicates the type of joint member.
PrimaryMember	Bit	Y	Indicates if the joint member is the primary member for the membership.
PLU	Char(20)	Y	The PLU of the item that the member was sold under. This field is only used for members that are sold separately from the core membership (such as add-on members).
Status	Integer	Y	<sup>2</sup> Status of the joint member. This is only set if the member status is different than the status of the membership.
OrderLineID	Integer	Y	Foreign key to OrderLines.OrderLineID
ContactGUID	UniqueIdentifier	Y	GUID associated to this contact. Used to uniquely identify this contact across any system.

### Indexes

Name	Kind	Columns	Purpose
PKJointMemberID	P	JointMemberID	Primary key
IJointMemberPassNo		PassNo	Speed up queries to load joint member records for a specified pass.
IJointMemberVisualID		VisualID	
IJointMemberContactID		ContactID	
IJointMembersOrderLineID		OrderLineID	
IJointMembersPreviousID		PreviousID	Index to allow searching for joint members based on the PreviousID field.

### <sup>1</sup> MemberType Values

Value	Gateway Constant Name	Description
0	MEMBER_TYPE_STANDARD	Normal joint member added as part of the membership.
1	MEMBER_TYPE_ADD_ON	Add-on member sold separately from the core membership.

### <sup>2</sup> Status Values

Value	Gateway Constant Name	Description
0	JOINT_MEMBER_VALID	Valid
1	JOINT_MEMBER_VOIDED	Voided
2	JOINT_MEMBER_RETURNED	Returned
3	JOINT_MEMBER_REMOVED	Removed
4	JOINT_MEMBER_SPLITOFF	Split-Off
5	JOINT_MEMBER_REPLACED	Replaced

## 16.2 PassEditFields

When editing a pass, each editable field is governed by a set of "rules". These rules are defined as part of the pass kind. The PassEditFields table, stores the rules for each of the fields that can be edited on a pass (by pass kind). There will be a row in the table for each rule, for each pass kind (the maximum value of FieldNumber will specify the number of rows per pass kind). Certain values in this table will be set by Galaxy and should never be changed. Others, as is the case with user-defined fields, are fully editable.

### Columns

Column	Type	Allow Nulls	Description
FieldID	Int	N	Primary key, always unique
FieldNumber	Int	N	Edit Field Number <sup>1</sup>
PassKind	Int	N	Foreign key to PassKinds.ID
DataRequired	Bit	Y	Is this a required field to have data in?
PrivRequired	Bit	Y	Does an agent require a special privilege to access this field?
ClearForNew	Bit	Y	Clear this field between passes?
FieldLength	Int	Y	Length of field. Sometimes not user definable.
DecimalPlaces	Int	Y	Number of decimal places <sup>2</sup>
FieldKind	Int	Y	Data type of field <sup>3</sup>
SwipeAction	Int	Y	Action to perform when a drivers license swipe is received <sup>4</sup>
FieldLabel	Varchar(100)	Y <sup>5</sup>	Label for the user to see while editing a pass
DefaultValue	Varchar(30)	Y	Default value for the field
MchoiceID	Int	Y	Reference to local BTree file SURVANS.DAT <sup>6</sup>
PrivRequiredForViewing	bit	Y	Determine if user requires a special privilege to view this field when upgrading, reissuing, renewing and viewing a pass. This does not apply to creating a new pass.
DisplayOnWebStore	Bit	Y	Determines if the Pass Edit Field of interest will be displayed on the web store.
Sequence	Int	Y	Determines the order in which the field is displayed. This sequencing is only supported on the Pass Kind Edit Screen and the Web Store. The standard Pass Edit Screen at appears in Galaxy POS does not honor this sequence column.
ClearOnRenewal	Bit	Y	Determines if the field will be cleared when renewing the pass. Only used for pass user fields.
ClearOnUpgrade	Bit	Y	Determines if the field will be cleared when upgrading the pass. Only used for pass user fields.
REAttributeTypeID	Int	Y	Attribute typed ID in The Raiser's Edge database that this value will map to
DisplayOnMemberPortal	Bit	Y	Determines if the Pass Edit Field will be displayed on the online member portal.
AllowEditInMemberPortal	Bit	Y	Determines if the Pass Edit Field can be edited on the online member portal.
PassEditFieldGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

Notes for PassEditFields table:

### <sup>1</sup> FieldNumber Values

Value	Description
0	First name
1	Middle initial / name
2	Last name
3	Street1
4	Street2
5	City
6	State
7	Zip / Postal code
8	Phone
9	Date of birth
10	Primary
11	Purchaser
12..19	User definable field 12 to 19
20	Max Uses
21	Max Party
22	User definable field 09
23	User definable field 10
24	Pass Holder Country Code
25	Pass Holder Email Address

### <sup>2</sup> DecimalPlaces value only used when FieldKind is 0 (Numeric)

### <sup>3</sup> FieldKind Values

Value	Description
0	Numeric data type
1	String data type
2	Logical data type
3	Multiple Choice data type
4	Raiser's Edge

### <sup>4</sup> SwipeAction Values

Value	Description
0	No action
1	Write data if none exists
2	Write data regardless of existing data

<sup>5</sup> Empty string values should be used when no label is desired such as the case with unused user definable fields

<sup>6</sup> Value will only be used if FieldKind value is 3 (Multiple Choice)

### 16.3 PassHistory

This table is used to store pass sales history including the initial sale, and all reissues, renewals and upgrades/downgrades. Each row in the table represents a type of "operation" on a pass. An "operation" can be defined as a pass sale, upgrade, downgrade, reissue, or renewal. One row will be inserted into this table for each "operation".

#### Columns

Column	Type	Allow Nulls	Description
PassHistoryID	Int	N	Primary key, always unique
PassID	Int	N	Foreign key to Passes.PassNo, specifying the pass record that this operation is for
TranNo	Int	N	The POS transaction number for this operation
NodeNo	Int	N	The POS node number that generated this operation
TranDate	DateTime	N	The date and time of this operation. (i.e., when the pass was sold, upgraded, reissued, etc).
PrevPrice	Money	N	The price of the pass before the operation
PrevTax	Money	N	The tax on the pass before the operation
Price	Money	N	The price of the pass after the operation
Tax	Money	N	The tax on the pass after the operation
PrevKind	Int	N	The pass kind of the pass before the operation
Kind	Int	N	The pass kind of the pass after the operation
TransactionType	Int	N	The type of transaction <sup>1</sup>
OrderID	Int	Y	Foreign key to the Orders table
PaymentContractID	Int	Y	Foreign key to the PaymentContracts table
UserID	NVarChar(60)	N	User name (User ID)
NoteID	Int	Y	ID of the pass history note

#### Indexes

Name	Kind	Columns	Purpose
PKPassHistoryID	P	PassHistoryID	Primary key - Unique ID in the table.
IXPassHistoryPassID		PassID	

<sup>1</sup> TransactionType

Value	Constant Name	Description
2	PASS_FKEY	A regular pass sale
6	REISSUE_PASS_FKEY	A pass reissue
7	RENEW_PASS_FKEY	A pass renewal
13	UPGRADE_PASS_FKEY	A pass upgrade / downgrade  The difference between an upgrade and downgrade will never be specified explicitly in this table. The system defines an upgrade history row as a row with a TransactionType value of 13 and a PrevPrice value less than or equal to the Price value. If the PrevPrice value is greater than the Price value, the system will <i>display</i> the history as a downgrade. An upgrade in a Gateway system is almost like exchanging an existing pass for another (different) pass; which is unlike a renewal / reissue where an existing pass is updated with the same information.
20	PH_TYPE_MEMBER_ADD_ON	
21	PH_TYPE_MEMBER_REMOVE	
22	PH_TYPE_MEMBER_RESTORE	
23	PH_TYPE_TICKET_UPGRADE	
24	PH_TYPE_MEMBER_SPLITOFF	
25	PH_TYPE_BACK_DATE_EXP	
26	PH_TYPE_EXTEND_EXP	
27	PH_TYPE_CHANGE_STATUS	
28	PH_TYPE_VOID_PASS	
29	PH_TYPE_PHOTO_ACCEPT	
30	PH_TYPE_PHOTO_REJECT	
31	PH_TYPE_UPDATE_CREDIT_CREDIT_CARD	
32	PH_TYPE_MAKE_PAYMENT	
33	PH_TYPE_FORCE_RECYLE	
34	PH_TYPE_UPDATE_ADDRESS	
35	PH_TYPE_PC_STATUS_CHANGE	
36	PH_TYPE_ADD_NOTE	
37	PH_TYPE_RETURN_PAYMENT	
38	PH_TYPE_PASS_KIND_CHANGE	Pass kind changed
39	PH_TYPE_USER_INFO	Pass holder photo or name information changed
40	PH_TYPE_DATE_INFO	Pass valid or expiration date changed
41	PH_TYPE_GENERAL_INFO	Pass information changed
42	PH_TYPE_UPGRADE_CREDIT	

## 16.4 PassHistoryRecordLog

This table is used to link PassHistory entries to GxRecordLog entries.

### Columns

Column	Type	Allow Nulls	Description
PassHistoryRecordLogID	Int	N	Primary key, always unique
PassHistoryID	Int	Y	Reference to PassHistory table
GxRecordLogID	Int	Y	Reference to GxRecordLog table

### Indexes

Name	Kind	Columns	Purpose
PKPassHistoryRecordLogID	P	PassHistoryRecordLogID	Primary key - Unique ID in the table.

## 16.5 PassPkgInstances

The PassPkgInstances contains links between a pass and packages that are owned by the pass. These links are made between the Passes and PkgInstances tables.

### Columns

Name	Type	Allow Nulls	Description
PassPkgInstanceID	Integer	N	Primary key, always unique.
PassID	Integer	N	Unique ID of the pass record. Foreign key to Passes.PassNo.
PkgInstanceID	Integer	N	Unique ID of the package. Foreign key to PkgInstances.PkgInstanceID.

### Indexes

Name	Kind	Columns	Purpose
PKPassPkgInstanceID	P	PassPkgInstanceID	Primary key - Unique ID in the table.
IXPassPkgInstPassID	A	PassID	Used for queries to find the packages owned by a pass.
IXPassPkgInstPkgInstanceID	A	PkgInstanceID	Used for queries to find the pass associated with a particular package.

## 16.6 Passes

The **Passes** table contains records for guest passes and the owners of those passes.

### Columns

Column	Type	Allow Nulls	Description
PassNo	Int	N	Primary key, always unique.
PassAcct	Char(24)	N	User definable alphanumeric account number.
First	Char(30)	Y	Pass holder's first name.
Last	Char(30)	Y	Pass holder's last name.
Middle	Char(30)	Y	Pass holder's middle name/initial.
Street1	Char(30)	Y	First line of address.
Street2	Char(30)	Y	Second line of address (often blank).
City	VarChar(40)	Y	Pass holder's city.
State	VarChar(40)	Y	Pass holder's state.
Zip	Char(10)	Y	Pass holder's zip / postal code.
CountryCode	Char(2)	Y	Pass holder country code. Foreign key reference to Countries.CountryCode that is an alternate key.
Phone	Char(16)	Y	Pass holder's phone number.
Email	Varchar(128)	Y	Pass holder email address
DateOpened	DateTime	Y	Date the pass was opened (first issued).
DateUsed	DateTime	Y	Last date the pass was used.
ValidDays	Int	Y	Number of days the pass is valid.
Kind	Int	Y	Foreign key to PassKinds.ID.
ValidUntil	DateTime	Y	The pass is valid until this day.
DOB	DateTime	Y	Passholder's date of birth.
Master	Int	Y	Link to another record in the Passes table - the primary pass holder for this pass.
Purchaser	Int	Y	Link to another record in the Passes table - the purchaser of this pass.
MaxParty	Int	Y	Max number of people that can be admitted with this pass.
Price	Money	Y	The original price of this pass.
Tax	Money	Y	The original amount of tax applied to this pass.
UseCount	Int	Y	The number of admissions this pass was used for.
ReissueCount	Int	Y	The number of times this pass was reissued.
Status	Int	Y	The current status of the pass. <sup>1</sup>
User01	VarChar(255)	Y	User definable field #1.
User02	VarChar(255)	Y	User definable field #2.
User03	VarChar(255)	Y	User definable field #3.
User04	VarChar(255)	Y	User definable field #4.
User05	VarChar(255)	Y	User definable field #5.
User06	VarChar(255)	Y	User definable field #6.
User07	VarChar(255)	Y	User definable field #7.
User08	VarChar(255)	Y	User definable field #8.
User09	VarChar(255)	Y	User definable field #9.
User10	VarChar(255)	Y	User definable field #10.
LimitCount	Int	Y	The number of times this pass was used to buy a pass-required ticket.
Company	Int	Y	Company this pass was sold from.
Category	Int	Y	Category number this pass belongs to.
SubCat	Int	Y	Sub-Category number this pass belongs to.
TaxFlags	Char(8)	Y	Taxes 1 to 8, the taxes applied when purchasing this pass (Y/N).
ProdNo	Int	Y	Product number from which this pass was sold.
Fkey	Int	Y	Function key number from which this pass was sold.
ValueKind	Int	Y	The meaning of the below Value field. <sup>2</sup>
Value	Money	Y	Value on the pass. <sup>3</sup>
AccessCode	Int	Y	Foreign key to AccessCodes.AccessCode.
VisualID	Varchar(40)	Y	Pass barcode.
FkeyFlags	Char(8)	Y	Function key options 1 to 8 (Y/N). <sup>4</sup>
FkeyKind	Int	Y	Function key kind. <sup>5</sup>
PriorPassAcct	Char(24)	Y	Pass account number prior to upgrade/downgrade, renewal or reissue.
PriorPassKind	Int	Y	PassKind.ID for the Pass kind prior to upgrade/downgrade, renewal or reissue.
NoteID	Int	Y	Foreign key to Notes.NoteID.
PictureID	Int	Y	Foreign key to Pictures.PictureID.
DiscountID	Int	Y	Reference to local BTree file DISCOUNT.DAT.
PLU	Char(20)	Y	This is the PLU for the ticket. This can either be in the form TICKETPPPLLFF for tickets associated with a product, or user-defined for tickets not associated with a product.
MaxUses	Int	Y	Maximum number of times the pass can be used.
TaxMethods	Char(8)	Y	The tax methods is an 8 character string, with each character being a '0', '1' or '2'. These eight characters refer to the eight possible taxes (similar to the Taxes column), and how each tax is applied to this ticket. <sup>6</sup>
NodeNo	Int	Y	Node number of POS that sold the pass

TransNo	Int	Y	Transaction number pass sold under Foreign key to CustContacts.CustContactID
AdultQty	Int	Y	Number of adult joint members on pass
ChildQty	Int	Y	Number of child joint members on pass
ValidFrom	DateTime	Y	The date a pass becomes valid
PendingPictureID	Int	Y	Foreign key to Pictures.PictureID for a picture that has is awaiting approval.
CustomerID	Int	Y	Holds the ID of the Customer that this pass is associated to. This is populated for passes with a PassKind where the CustomerPass option is true.
GalaxySiteID	Int	Y	Foreign Key reference to Sites.GalaxySiteID
OrderID	Int	Y	When a pass is sold as part of an order, this value is a foreign key to Orders.OrderID.
DebitCardID	Int	Y	Unique ID of the debit card record associated with the pass. This value is populated when a package containing stored value is sold with the pass. Foreign key to DebitCards.
LastExpiration	DateTime	Y	Set to the previous expiration date when a pass is renewed AND the pass kind of the renewal is configured to calculate the new expiration date based on number of days from first use and has its FirstUseExpireCalculationMode column set to 1 (FIRST_USE_EXPIRE_MODE_FIRST_USE_IN_RENEWAL_PERIOD). Used by ACS to properly calculate the new expiration date when the pass if first used after a renewal.
PurchaserContactID	Int	Y	Link to the contact that purchased this pass. Foreign key to CustContacts.CustContactID. Replaces the Purchaser column, which linked to a pass instead of a contact.
SendPassTo	Int	Y	Indicates the contact to which the pass should be sent (either the passholder or purchaser). <sup>7</sup>
Reprinted	Bit	Y	Indicates that the pass has already been reprinted.
ReprintOnNextScan	Bit	Y	Indicates that the pass should be reprinted on next scan at ACS.
MustUseByDate	DateTime	Y	The date the pass must be used by.
MustUseByState	Int	Y	Indicates the must use by state. <sup>8</sup>

**Indexes**

Name	Kind	Columns	Purpose
PKPassesPassNo	P	PassNo	Primary Key.
IXPassesPassAcct		PassAcct	Used to find a pass by its account.
IXPassesVisualID		VisualID	Used to find a pass by its barcode.
IXPassesContactID		ContactID	Used for load passes info if OE contact has a Pass associated with it.
IXPassesKind		Kind	Optimizing pass search queries
IXPassesPurchaserContactIDKind		Purchaser, ContactID, Kind	To eliminate Pass Purchaser and Primary Query Timeouts
IXPassesCustomerID		CustomerID	Used when querying passes by CustomerID
IXPassesPriorPassAcct		PriorPassAcct	Index to allow searching for passes based on the PriorPassAcct field.
IXPassesMaster		Master	Searching by master field
IXPassesPurchaserContactID		PurchaserContactID	Used for loading passes by purchaser contact
IXPassesStatusOrderID		Status, OrderID	Used when searching for orders that are filtered by status
IXPassesDebitCardID		DebitCardID	Index to allow searching for the pass that a debit card record belongs to
IXPassesFirst		First	
IXPassesLast		Last	

**<sup>1</sup> Status Values**

Value	Gateway Constant Name	Description
0	PASS_VALID	Pass is valid.
1	PASS_VOIDED	Pass has been voided.
2	PASS_RETURNED	Pass has been returned.
3	PASS_REPLACE	Pass was replaced (from upgrade, reissue, renewal) or Pass was reprinted using the Reprint function in Galaxy
4	PASS_PURCHASER	This is a purchaser pass.
5	PASS_EXPIRED	Pass is expired.
6	PASS_UPGRADED	Pass was upgraded using the Ticket Upgrade function in Galaxy.
7	PASS_REPRINTED	Pass was reprinted using the Reprint function in Galaxy, and the "Create New VisualID When Reprinting Tickets" option in Order Entry config is enabled. SuperTicket chain validation is aborted when any ticket with this status is found in the chain.
8	PASS_BLOCKED	Pass is blocked.
9	PASS_UNISSUED	Pass was created as part of an order, but has not yet been issued.
10	PASS_RESTRICTED_ADMISSION	Pass is valid for admission, but is not valid for pass-required tickets or pass-required discounts.
11	PASS_APPROVED	This status corresponds to a pass photo status of Approved when config option is selected to set pass and pass photo statuses to be the same.
12	PASS_NOT REVIEWED	This status corresponds to a pass photo status of Not_Reviewed when config option is selected to set pass and pass photo statuses to be the same.
13	PASS_REJECTED	This status corresponds to a pass photo status of Rejected when config option is selected to set pass and pass photo statuses to be the same.
14	PASS_NOT PRINTED	The pass exists and IDs have been generated, but it was not printed. This is used for SIAE in Order Entry since pass information must be generated at the time of payment.
15	PASS_INACTIVE	Pass is inactive.

**<sup>2</sup> ValueKind Values**

Value	Gateway Constant Name	Description
0	kdCurrency	The "Value" field refers to a monetary value (currency).
1	kdPoints	The "Value" field refers to a points value.

<sup>3</sup> The value associated with the "Value" field is determined by the ValueKind field. See previous reference (<sup>2</sup>)<sup>4</sup> FkeyFlag values will be a series of eight characters each with a value of "Y" or "N" corresponding to options that were enabled/disabled at the time the pass was sold. The following chart refers to the positions of the Y/N characters in the series of eight. For example, position four would be the fourth Y/N in the series of eight Y/N characters in the FkeyFlag field. If the value is Y the option in the description column as enabled at the time of the pass sale.

Position	Description
1	The same pass kind is kept when reissuing a pass.
2	The pass expiration is reset when reissuing a pass.
3	Three ticket entries are journalized instead of one when reissuing passes.
4	When returning a pass, it should be selected from a list instead of typing in a barcode.
5	The old pass ID number is kept on reissues, regardless of pass kind settings.
6	Reserved, not currently used.
7	Reserved, not currently used.
8	Reserved, not currently used.

**5 FkeyKind Values**

Value	Description
2	Pass function key.
6	Reissue function key.
7	Renewal function key.
13	Upgrade function key.

**6 TaxMethod values, per character**

Value	Gateway Constant Name	Description
"0"	ITEM_TAX_PER_ITEM	The tax was applied on a per item basis.
"1"	ITEM_TAX_PER_TRANS	The tax was applied on a per transaction basis.
"2"	ITEM_TAX_AUTO	The tax follows the tax method settings defined in the system tax configuration, which could vary.
Blank or NULL	N/A	It is assumed that all taxes were applied on a per item basis.

**7 SendPassTo Values**

Value	Gateway Constant Name	Description
0	SEND_PASS_TO_PASSEHOLDER	The pass should be sent to the passholder.
1	SEND_PASS_TO_PURCHASER	The pass should be sent to the purchaser.

**8 MustUseByState Values**

Value	Gateway Constant Name	Description
0	PASS_MUST_USE_BY_NOT_SET	The MustUseByDate has not been set or is not applicable to the pass.
1	PASS_MUST_USE_WAITING	The MustUseByDate has been set and the Pass has not been used.
2	PASS_MUST_USE_VALID	The pass was used before the MustUseByDate.
3	PASS_MUST_USE_LAPSED	The MustUseByDate has lapsed before the pass was used.

## 16.7 PassKinds

The **PassKind** table contains definitions for different types of passes. Typically, the pass kind specifies how long a pass is valid, but may also be used to specify information about account numbers, column requirements, and column types.

### Columns

Column	Type	Allow Nulls	Description
ID	Int	N	Primary key, always unique
ExternalPassKind	Int	Y	User-definable pass kind number
Name	Char(30)	Y	Pass kind name
ValidDays	Int	Y	Number of days the pass will be valid for. <sup>1</sup>
ValidMonths	Int	Y	Used when calculating an expiration date for a pass. Number of months from pass sale.
ValidYears	Int	Y	Used when calculating an expiration date for a pass. Number of years from pass sale.
ValidUntil	DateTime	Y	The date the pass will be valid until. <sup>2</sup>
ExpireType	Int	Y	Type of expiration. <sup>3</sup>
ExpireOverride	Bit	Y	Allow ticket seller to edit expiration date of a pass being sold?
UseSystemID	Bit	Y	Have the system provide all pass account numbers?
NewIDOnReissue	Bit	Y	[Replaced by ReissueUpdateType]
NewIDOnRenew	Bit	Y	[Replaced by RenewUpdateType]
PassRequiredLimit	Int	Y	Number of times a pass can be used towards a pass-required ticket
BlankPass	Bit	Y	Column value not used in Galaxy
NewIDOnUpgrade	Bit	Y	[Replaced by UpgradeUpdateType]
ContractType	Int	Y	Does this pass require a contract to be printed before selling? If so, what type of contract will be used. <sup>6</sup>
ContractPrinterNo	Int	Y	Printer to print the contract ticket set to. <sup>4</sup>
ContractTktSet	Int	Y	Contract ticket set used. <sup>4</sup>
User01	Char(20)	Y	Answer to PASSKIND_FIELD1 keyword
User02	Char(20)	Y	Answer to PASSKIND_FIELD2 keyword
User03	Char(20)	Y	Answer to PASSKIND_FIELD3 keyword
User04	Char(20)	Y	Answer to PASSKIND_FIELD4 keyword
User05	Char(20)	Y	Answer to PASSKIND_FIELD5 keyword
User06	Char(20)	Y	Answer to PASSKIND_FIELD6 keyword
User07	Char(20)	Y	Answer to PASSKIND_FIELD7 keyword
User08	Char(20)	Y	Answer to PASSKIND_FIELD8 keyword
User09	Char(20)	Y	Answer to PASSKIND_FIELD9 keyword
User10	Char(20)	Y	Answer to PASSKIND_FIELD10 keyword
JournalizeNotes	Bit	Y	Make journal entries for pass notes when selling
ResetUsesOnRenew	Bit	Y	Reset pass use count when renewing?
DiscountID	Int	Y	Reference to local BTree file DISCOUNT.DAT
PhotoColors	Int	Y	Number of colors used to store a pass photo
PhotoFormat	Int	Y	Pass photo format. <sup>5</sup>
ResetUsesPeriodically	Bit	Y	Controls whether UseCount field is reset periodically by the ResetPassUsage program
ResetUsesFrequency	Int	Y	How often to reset the usages, in this format:  (years * 10000) + (months * 100) + (days)
NextUsesReset	DateTime	Y	Date number of usages will be reset
ResetMaxUses	Bit	Y	If 1, reset MaxUses to AccessCodes.MaxUses  If 0, do not reset MaxUses
RenewFromDays	Int	Y	Defines the number of days a pass can be renewed to, before it expires
RenewThruDays	Int	Y	Defines the number of days a pass can be renewed to, after it expires
DefaultPassRequired	Bit	Y	Allow passes of this kind to become associated with subsequent tickets and items
JointMembership	Bit	Y	Indicates if this Pass Kind uses Joint memberships.
MaxAdults	Int	Y	Indicates the maximum number of adults that can be added to a membership/pass.
MaxChildren	Int	Y	Indicates the maximum number of children that can be added to a membership/pass. -1 denotes unlimited.
MaxGuests	Int	Y	Indicates the maximum number of guests that can be added to a membership/pass.
CardIDAssignment	Int	Y	Indicates how ID numbers will be assigned to each card printed. <sup>7</sup>
CardProduction	Int	Y	Indicates the rule for printing membership cards. <sup>8</sup>
DuplicateCardQty	Int	Y	Indicates the number of cards to print for a given membership.
RequireChildDOB	Bit	Y	Indicates if child joint members must provide their date of birth.
RequireAdultDOB	Bit	Y	Indicates if adult joint members must provide their date of birth.
LimitAccess	Bit	Y	Determines if passes of this pass kind will require a privilege to modify.
CustomerPass	Bit	Y	Determines if the passkind saves the CustomerID on the pass record to associate the pass to a customer.
MinimumAge	Int	Y	The minimum age of a Contact to be eligible to purchase the Pass, or the lower value of an age range.
MaximumAge	Int	Y	The maximum age of a Contact to be eligible to purchase the Pass, or the highest value for an age range.
MinQtyNamedAdults	Int	Y	For Joint Memberships, setting on the PassKind to indicate the number of Adult members whose names must be added when the membership is sold.
ReissueUpdateType	Int	Y	Indicates the method that will be used to update the pass when a reissue is performed. It determines if there will be a new pass record created, and the rules for updating visual IDs. This replaces the NewIDOnReissue column, since there are now more than two options. <sup>9</sup>
RenewUpdateType	Int	Y	Indicates the method that will be used to update the pass when a renewal is performed. It determines if there will be a new pass record

			created, and the rules for updating visual IDs. This replaces the NewIDOnRenew column, since there are now more than two options. <sup>9</sup>
UpgradeUpdateType	Int	Y	Indicates the method that will be used to update the pass when an upgrade is performed. It determines if there will be a new pass record created, and the rules for updating visual IDs. This replaces the NewIDOnUpgrade column, since there are now more than two options. <sup>9</sup>
RestrictJointChildAge	Bit	Y	Determines if the system will enforce a maximum age for a joint member to be considered a child. This setting overrides the child age restriction option in general configuration.
JointChildMaxAge	Int	Y	The maximum age for a joint member to be considered a child. This overrides the maximum child age setting in general configuration.
PostalCodeRangeID	Int	Y	Allows for geographically limiting use of a pass kind; FK to PostalCodeRanges.PostalCodeRangeID
MaxAddons	Integer	Y	Maximum number of Add-On products that can be sold with a membership
JMPictureCaptureMethod	Integer	Y	<sup>10</sup> Indicates the method of capturing pictures for a joint membership at time of sale.
LifetimeCard	Bit	Y	Indicates that the system will use the same visual ID for the lifetime of the pass across all renewals and upgrades.
NewVisualIDForJMRenew	Bit	Y	Indicates that a new visual ID will be created for the joint member cards when the membership is renewed.
NewVisualIDForJMUupgrade	Bit	Y	Indicates that a new visual ID will be created for the joint member cards when the membership is upgraded.
FirstUseExpireCalculationMode	Integer	Y	Determines how the expiration date is calculated when the pass is renewed. If set to 0, the expiration is calculated based on renewal date and is set on the pass at the time of renewal (default legacy logic). If set to 1, the expiration date on the renewed pass remains null until it is used, at which point ACS calculates the expiration date based on either the original expiration date (Passes.LastExpiration) or the current date, whichever is greater. <sup>11</sup>
NewVisualIDForJMRissue	Bit	Y	Indicates that a new visual ID will be created for the joint member cards when the membership is reissued.
CreateDualRelationships	Bit	Y	Indicates if dual relationships will be created when multiple passes are sold in the same transaction or order.
SystemIdMethod	Int	Y	<sup>12</sup> Indicates the method used when generating a system account number.
MatchEntitlementAddOnDates	Bit	Y	Indicates if entitlement add-on should copy start and expiration dates of the pass.
Inactive	Bit	Y	Indicates whether or not a pass kind is inactive.
PassKindGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems
MustUseByDays	Int	Y	The number of days after the pass has been issued that the user has to activate the pass.
MustUseByMonths	Int	Y	The number of months after the pass has been issued that the user has to activate the pass.
MustUseByDate	DateTime	Y	The date after a pass has been issued that the user has to activate the pass.
MustUseBasis	Int	Y	The basis for computing the date by which the pass must be used to activate the pass. <sup>13</sup>

**Indexes**

Name	Kind	Columns	Purpose
PKPassKindsID	P	ID	Primary Key.
AKPassKindsExternalPassKind	A	ExternalPassKind	Used to find a pass kind by its user-definable pass kind number.

<sup>1</sup> Only used if ExpireType is 0, 3, or 4.<sup>2</sup> Only used if ExpireType is 1**<sup>3</sup> ExpireType Values**

Value	Description
0	Pass will expire in a set number of days from the date the pass is issued
1	Pass will expire on a set date
2	Pass will expire in a set number of days from the first use
3	Pass will expire based on a specific interval from the date the pass is issued to the end of the month
4	Pass will expire in a set number of days from the pass sale date

<sup>4</sup> Only used if the ContractType value is non-zero.**<sup>5</sup> PictureFormat Values**

Value	Description
0	Bitmap
1	JPEG
2	PCX

**<sup>6</sup> ContractType Values**

Value	Description
0	None - No contract is required when selling a pass with this kind
1	By Pass - Each individual pass requires a contract. The contract for each pass must be printed before leaving the pass data entry window.
2	By Transaction - One contract is required for all passes sold in a transaction.

**<sup>7</sup> CardIDAssignment Values**

Value	Gateway Constant Name	Description
0	SAME_ID_PER_MEMBER	Each membership card has same ID
1	UNIQUE_CARD_PER_MEMBER	Each membership card has a unique ID

**<sup>8</sup> CardProduction Values**

Value	Gateway Constant Name	Description
0	GENERATE_SINGLE_CARD	Generate a single card for entire joint membership
1	GENERATE_CARD_PER_MEMBER	Generate a card for every named member

2	GENERATE_CARD_PER_ADULT	Generate a card for every named adult
---	-------------------------	---------------------------------------

**9 UpdateType Values**

Value	Gateway Constant Name	Description
0	PASS_UPDATE_TYPE_SAME_RECORD	When the pass is reissued, renewed, or upgraded, the same pass record is used. The pass account number does not change, and the visual ID stays the same. This equivalent to having the UseNewIDOn... option turned off.
1	PASS_UPDATE_TYPE_NEW_RECORD	When the pass is reissued, renewed, or upgraded, a new pass record is generated and the old one becomes invalid. The new record has a new pass account number and visual ID. This is equivalent to having the UseNewIDOn... option turned on.
2	PASS_UPDATE_TYPE_NEW_VISUAL_ID	When the pass is reissued, renewed, or upgraded, the same pass record is used, but that record is given a new visual ID. The pass account number stays the same. A SuperTicket record is generated with the old visual ID, so it is not completely erased from the system.

**10 JMPictureCaptureMethod Values**

Value	Gateway Constant Name	Description
0	JM_PICTURE_CAPTURE_METHOD_ITEM	Capture pictures for the primary only, using the rules from the item record.
1	JM_PICTURE_CAPTURE_METHOD_NAMED_MEMBERS	Capture pictures for all named members in the membership
2	JM_PICTURE_CAPTURE_METHOD_ADULT_MEMBERS	Capture pictures for all adult members in the membership

**11 FirstUseExpireCalculationMode Values**

Value	Gateway Constant Name	Description
0	FIRST_USE_EXPIRE_MODE_RENEWAL_DATE	Expiration date calculated based on renewal date.
1	FIRST_USE_EXPIRE_MODE_FIRST_USE_IN_RENEWAL_PERIOD	Expiration date calculated on first use within the renewal period. If the original expiration date is greater than the first use date, the original expiration date is used.

**12 SystemIdMethod Values**

Value	Gateway Constant Name	Description
0	TPassAccountGenerationMethod.SERIAL	Use the Serial portion of the Visual ID.
1	TPassAccountGenerationMethod.GLOBAL	Use the SQL Sequence "SequentialPassCounterSequence".

**13 MustUseBasis Values**

Value	Gateway Constant Name	Description
0	MUST_USE_BASIS_NONE	No limitation on when the pass must be used.
1	MUST_USE_BASIS_RELATIVE_TO_ISSUE	Require use of the pass based on the number and days and months after issuance.
2	MUST_USE_BASIS_ABSOLUTE_DATE	Require use of the pass by an absolute date.

## 16.8 PassKindGroups

The **PassKindGroups** table facilitates the grouping of PassKinds together for ACS reservation purposes.

### Columns

Column	Type	Allow Nulls	Description
PassKindGroupID	Int	N	Primary key, always unique.
PassKindGroupGUID	UniqueIdentifier	N	GUID associated to this PassKindGroup. Used to uniquely identify this PassKindGroup record across any system.
Name	Char(50)	N	Pass Kind Group name.
Description	Char(250)	Y	Pass Kind Group description.
AttributeValueGroupID	Int	Y	Foreign key reference to AttributeValues.AttributeValueGroupID. Generated from the GatewayCounters table and will be unique for a group of values.
Inactive	Bit	N	True if item is Inactive, and not visible in most picklists.

### Indexes

Name	Kind	Columns	Purpose
PKPassKindGroupID	P	PassKindGroupID	Primary Key.
IXPassKindGroupGUID	A	PassKindGroupGUID	Uniquely identifies this PassKindGroup across any system.

## 16.9 PassKindGroupDetails

The **PassKindGroupDetails** table defines the association of PassKinds to PassKindGroups for ACS reservation purposes.

### Columns

Column	Type	Allow Nulls	Description
PassKindGroupDetailID	Int	N	Primary key, always unique.
PassKindGroupDetailGUID	UniqueIdentifier	N	GUID associated to this PassKindGroupDetail. Used to uniquely identify this PassKindGroupDetail record across any system.
PassKindGroupGUID	UniqueIdentifier	N	Foreign key reference to PassKindGroups.PassKindGroupGUID
PassKindID	Int	N	Foreign key reference to PassKinds.ID

### Indexes

Name	Kind	Columns	Purpose
PKPassKindGroupDetailID	P	PassKindGroupDetailID	Primary Key.
IXPassKindGroupDetailGUID	A	PassKindGroupDetailGUID	Uniquely identifies this PassKindGroupDetail record across any system.
IXPassKindGroupGUID	F	PassKindGroupGUID	Foreign key reference to PassKindGroups.PassKindGroupGUID
IXPassKindID	F	PassKindID	Foreign key reference to PassKinds.ID

## 16.10 PassKindsRE

This table links the Raiser's Edge member categories with Gateway Passkinds.

### Columns

Name	Type	Allow Nulls	Description
PassKindsREID	Integer	N	Primary key, always unique
PassKindID	Integer	N	Foreign key reference to PassKinds.ID
MemberCategoryID	Integer	Y	Member category ID from Raiser's Edge software
AccessCode	Integer	Y	Access Code
PLU	Char(20)	Y	Pass PLU
SubCategoryID	Integer	Y	Member subcategory ID from the Raiser's Edge software

### Indexes

Name	Kind	Columns	Purpose
PKPassKindsREPassKindsREID	P	PassKindREID	Primary key.
IXPassKindsREPassKindID		PassKindID	Main queries will be by PassKindID

### 16.11 PassReqLog

The PassReqLog table contains a log of passes used for pass-required tickets. One row will be inserted into this table each time a pass-required ticket is used. Note, this table will only be updated if the Galaxy "Log Pass Required Tickets" general configuration option is enabled.

#### Columns

Column	Type	Allow Nulls	Description
PassReqLogID	Int	N	Primary key, always unique
PassID	Int	N	Foreign key to Passes.PassNo
VisualID	VarChar(40)	Y	Foreign key to Tickets.VisualID
TransDate	DateTime	N	Transaction date / time
PLU	Char(20)	Y	Foreign key to Items.PLU, this is the PLU for the ticket. This can either be in the form TICKETPPPLLFF for tickets associated with a product, or user-defined for tickets not associated with a product.
Price	Money	Y	Price paid for ticket/item after subtracting discounts and excluding taxes. Matches the price in JnlTickets.Price.
Tax	Money	Y	Total amount of tax paid with ticket.
TransactionNumber	Integer	Y	Transaction number this ticket/item belongs to.
NodeNo	Integer	Y	Node number of the POS generating the transaction.
OrderLineID	Integer	Y	Holds the OrderLineID of the line that this record was created for. This will help in maintaining the proper limits for unissued pass required tickets that are in orders.

#### Indexes

Name	Kind	Columns	Purpose
PKPassReqLogPassReqLogID	P	PassReqLogID	Primary Key.
IXTransDatePassIDPLU		TransDate, PassID, PLU	Locate Pass required tickets to be retrieved by date range.
IXVisualIDOrderLineID		VisualID,OrderLineID	Improve query performance
IXPassID		PassID	Improve query performance

## 16.12 PassRestrictions

Contains restrictions for using pass-required tickets for by pass kind product-based tickets when the "Restrict by Pass Kind" general configuration option is enabled.

### Columns

Column	Type	Allow Nulls	Description
PassRestrictionID	Int	N	Primary key, always unique
RestrictionType	Int	Y	The type of the restriction <sup>1</sup>
ProdNo	Int	Y	Product Number
LevelNo	Int	Y	Level Number
FkeyNo	Int	Y	Fkey Number
PassKindID	Int	Y	The ID of the Pass Kind used
PLU	Char(20)	Yes	Reference to the item PLU (Items.PLU) to which this restriction applies
PassRestrictionGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKPassRestrictionsRestrictionID	P	PassRestrictionID	Primary Key.
AKPassRestrictionsTypeKindPLU	A	RestrictionType, PassKindID, PLU	Alternate key

<sup>1</sup> **RestrictionType** Values

Value	Description
1	Ticket
2	Discount (currently not used)

**17 Photo**

## 17.1 Pictures

The Pictures table contains pictures captured by the system, as is the case with picture passes.

### Columns

Column	Type	Allow Nulls	Description
PictureID	Int	N	Primary key, always unique
Origin	Int	Y	Format of ImageData <sup>1</sup>
PixelDepth	Int	Y	Number of colors ImageData is stored in
ImageData	Image	Y	Bytes on an image
Status <sup>2</sup>	Int	Y	Determines the current state of the picture.
BaseID	Int	Y	This allows for grouping of pictures. If this is the first picture in a chain of pictures the BaseID will be equal to the PictureID. All other associated pictures will have a BaseID equal to the BaseID of the first picture.
PictureGUID	UniqueIdentifier	Y	GUID associated to this picture. Used to uniquely identify this picture across any system.

### Indexes

Name	Kind	Columns	Purpose
PKPicturesPictureID	P	PictureID	Primary key
IXPicturesBaseID	IX	BaseID	To improve performance of queries referencing the BaseID
IXPictureGUID	IX	PictureGUID	To improve performance of queries referencing the PictureGUID

Notes for Picture table:

### <sup>1</sup> PictureFormat Values

Value	Description
0	Bitmap
1	JPEG
2	PCX

### <sup>2</sup> Status Values

Value	Gateway Constant Name	Description
0	STATUS_VERIFIED	This visually verified picture is now the current picture in a group of pictures defined by the BaseID column. Pictures attached to passes sold via POS or OE are automatically verified.
1	STATUS_PENDING	This picture is currently awaiting verification.
2	STATUS_REJECTED	This denied picture will not be used.
3	STATUS_APPROVED	This picture has been approved via the photo review manager.

## 18 Resource Management

The Galaxy Resource Management module provides a means to manage the use of resources within your facility. The module provides two different functions governing resources.

Resources can be defined to assign a set number of tickets to be available for sale. The resources are then associated with events. These events are predetermined time slots for use of the resource. Associating resources with events allows control over the number of tickets that can be sold for a single time slot. Integrated reports are provided to give indications of revenue for each event as well as the capacity amounts for each event.

Resources can also be defined for reserving areas, personnel, or equipment for a user defined time slot. This definition allows control over scheduling and resource availability. Reports provide a print out of current reservations for scheduling purposes.

## 18.1 EventManagementActivity

The EventManagementActivity table records all activity that modifies the state inside the Event Management Service.

### Columns

Column	Type	Allow Nulls	Description
EventManagementActivityID	Identity	N	Primary key, always unique
ActivityKind	int	N	Identifies the type of activity
ActivityDateTime	datetime	N	The date and time the activity occurred
HttpRequestID	uniqueidentifier	Y	A unique identifier for the request that initiated the activity
SessionID	uniqueidentifier	N	A foreign key reference to the EventManagementSessions table
SessionStatus	int	N	The current status of the session
UserID	int	N	A foreign key reference to the GxUsers table
AppSourceID	int	Y	A foreign key reference to the AppSources table
IPAddress	nvarchar(100)	Y	The IP Address of the machine that sent the request
ExpirationDateTime	datetime	Y	The date and time that the session expires
EventID	int	Y	A foreign key reference to the RMEvents table
NoteID	int	Y	A foreign key reference to the Notes table
ContactID	int	Y	The contact associated with the EventManagementActivity. Foreign key to CustContacts table.

### Indexes

Name	Kind	Columns	Purpose
PKEventManagementActivityID	P	EventManagementActivityID	Primary Key.

## 18.2 EventManagementSeatActivity

This table records all activity that modifies the state of a reserved seat inside the Event Management Service.

### Columns

Column	Type	Allow Nulls	Description
EventManagementSeatActivityID	Integer	N	Primary key
EventManagementActivityID	Integer	N	A foreign key reference to the EventManagementActivity table
EventID	Integer	N	A foreign key reference to the RMEvents table
RSEventSeatMapID	Integer	N	A foreign key reference to the RSEventSeatMaps table
RSEventSeatID	Integer	N	A foreign key reference to the RSEventSeats table
RSSeatID	Integer	Y	A foreign key reference to the RSSeats table
RSEventPriceID	Integer	Y	A foreign key reference to the RSEventPrices table
RSPriceID	Integer	Y	A foreign key reference to the RSPrices table
RSHoldCodeID	Integer	Y	A foreign key reference to the RSHoldCodes table
SeatStatus	Integer	N	The status of the seat
SeatStatusChangeType	Integer	N	Signifies how a seat's status changed
SectionName	Nvarchar(20)	N	The name of the section for this seat
RowName	nvarchar(20)	N	The name of the row for this seat
SeatName	nvarchar(20)	N	The name of the seat for this seat
CoordX	Integer	N	The X-axis coordinate for the seat
CoordY	Integer	N	The Y-axis coordinate for the seat
Rank	Integer	N	The seat's rank
Adjacency	Integer	N	The seat's adjacency
DateTimeReserved	datetime	Y	The date and time the seat was reserved
VisualID	nvarcha	Y	The Visual ID currently associated to the seat
ContactID	Integer	Y	A foreign key reference to the CustContacts table
SessionGUID	uniquei	Y	A foreign key reference to the EventManagementSessions table

### Indexes

Name	Kind	Columns	Purpose
PKEventManagementSeatActivityID	P	EventManagementSeatActivityID	Primary Key.

### <sup>1</sup> SeatStatusChangeTypes Values

Value	Gateway Constant Name	Description
0	NoChange	No change in the seat's status
1	AvailableToReserved	The seat's status changed from available to reserved
2	AvailableToSold	The seat's status changed from available to sold
3	AvailableToHold	The seat's status changed from available to hold
4	ReservedToAvailable	The seat's status changed from reserved to available
5	ReservedToSold	The seat's status changed from reserved to sold
6	ReservedToHold	The seat's status changed from reserved to hold
7	SoldToAvailable	The seat's status changed from sold to available
8	SoldToReserved	The seat's status changed from sold to reserved
9	SoldToHold	The seat's status changed from sold to hold
10	HoldToAvailable	The seat's status changed from hold to available
11	HoldToReserved	The seat's status changed from hold to reserved
12	HoldToSold	The seat's status changed from hold to sold

### 18.3 EventManagementSessions

#### Columns

Column	Type	Allow Nulls	Description
EventManagementSessionID	Uniqueidentifier	N	Primary key, always unique
Status	Int	N	
UserID	Int	N	
AppSourceID	DateTime	Y	
LastActivityAt	DateTime	Y	
ExpirationDateTime	Nvarchar(100)	Y	
IPAddress	Int	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKEventManagementSessionID	P	EventManagementSessionID	Primary Key.
IXEventManagementSessionsAppSourceID	IX	AppSourceID	
IXEventManagementSessionsUserID	IX	UserID	

## 18.4 EventSaleLimits

The EventSaleLimits Table contains Sales Capacity Limits by Sales Channel for Events.

### Columns

Column	Type	Allow Nulls	Description
EventSaleLimitID	Int	N	Primary key, always unique
EventID	Int	N	The Unique ID of the associated Event
SalesChannelID	Int	N	The Unique ID of the associated Sales Channel
OnSale	DateTime	N	Start Date & Time the Event can be sold
OffSale	DateTime	N	Stop Date & Time the Event can be sold
CapacityUnit	Int	N	Format of the capacity field (0 = PERCENTAGE, 1 = QUANTITY)
Capacity	Int	N	Capacity that can be sold for this Sales Channel

### Indexes

Name	Kind	Columns	Purpose
PKEventSaleLimitID	P	EventSaleLimitID	Primary Key.
IXEventSaleLimits		EventID, SalesChannelID	Load all data for a given event and sales channel

### <sup>1</sup> CapacityUnit Values

Value	Gateway Constant Name	Description
0	PERCENTAGE_CAPACITY_UNIT	Value in Capacity column is the percentage of the capacity limit
1	QUANTITY_CAPACITY_UNIT	Value in Capacity column is the actual capacity limit in the number

## 18.5 HoldDetails

Hold Set Details are records of the PLUs and/or the ItemGroups included in a Hold Set or Hold Set Template.

### Columns

Column	Type	Allow Nulls	Description
HoldDetailID	Integer	No	Primary key, always unique
DetailType	Integer	No	A numeric code identifying the type of HoldDetail. <sup>1</sup>
AuxID	Integer	No	RMCapacity.HoldGroupID or EventTypeID
PLU	Char(20)	Yes	PLU of tickets to be held (allow nulls as detail may have ItemGroupID OR PLU, not both)
ItemGroupID	Integer	Yes	ItemGroupID for item group to be held (allow nulls as detail may have ItemGroupID OR PLU, not both)

### Indexes

Name	Kind	Columns	Purpose
PKHoldDetailsHoldDetailID	P	HoldDetailID	Primary Key
IXHoldDetailsDetailTypeAuxID		DetailType, AuxID	Used to search for details by DetailType and AuxID

### <sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
0	EVENT_TYPE_HOLD	The detail belongs to a Hold Template for an Event Type
1	EVENT_HOLD	The detail belongs to a RMCapacity record for an Event

## 18.6 HoldTemplates

Hold Templates are definitions of groups of tickets that are ordinarily held back from general sales until or unless released.

### Columns

Column	Type	Allow Nulls	Description
HoldTemplateID	Integer	No	Primary key, always unique
HoldName	VarChar(50)	No	Hold Template name
Description	VarChar(255)	Yes	Description of hold template
EventTypeID	Integer	No	ID of the Event Type or Resource to which this hold is applied based on the HoldType
TotalHoldQuantity	Integer	No	Total tickets held for all details of this hold template
AllowRelease	Bit	No	Indicates whether the tickets held can be released
HoldType	Integer	Yes	Indicates the type of hold <sup>1</sup>

### <sup>1</sup> Hold Type Values

Value	Gateway Constant Name	Description
0	EVENT_TYPE_HOLD	Hold Template defined on EventType. The EventTypeID column will contain the EventTypeID from the RMEventTypes table
2	RESOURCE_HOLD	Hold template defined on a Resource. The EventTypeID column will contain the ResourceID from the RMResources table

### Indexes

Name	Kind	Columns	Purpose
PKHoldTemplatesHoldTemplateID	P	HoldTemplateID	Primary key
IXHoldTemplatesEventTypeID		EventTypeID	Used to search for templates by EventTypeID

## 18.7 RentalSerial

The RentalSerial table links a rental (with serial ID and resource ID) to a Visual ID (ticket #). Each rental then has a status, which will change depending on what is done with the rental.

### Columns

Column	Type	Allow Nulls	Description
RentalSerialUniqueID	Int	N	Primary key, always unique.
RentalSerialID	Int	N	Unique serial number for each rental.
VisualID	VarChar(40)	N	VisualID of ticket associated with the rental.
ResourceID	Int	N	Foreign key to RMResources.ResourceID, representing the resource that is the rental. Must contain a value greater than zero, and point to a valid resource.
ReservationID	Int	N	Foreign key to RMReservations.ReservationID. Represents the reservation record for the rental. Must contain a value greater than zero, and point to a valid reservation.
Status	Int	N	Code representing the status of the rental. <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKRentalSerialUniqueID	P	RentalSerialUniqueID	Primary key, always unique (Index name does not conform to standards due to 30 character name limit in Oracle).
IXRentalSerialVisualID		VisualID	Used to search for rentals by VisualID.
IXRentalSerialIDStatus		RentalSerialID, Status	Used to load a unique rental, where the RentalSerialID's match and the Status of the record is zero (Rental checked out). (Index name does not conform to standards due to 30 character name limit in Oracle).

<sup>1</sup> Status Values

Value	Description
0	Rental checked out.
1	Rental checked in.
2	Rental returned.
3	Rental exchanged.
4	Rental voided.

## 18.8 RMAudioTypes

The RMAudioTypes table contains the information about the different sound formats that are used in a film.

### Columns

Column	Type	Allow Nulls	Description
AudioTypeID	Int	N	Primary key, always unique.
Abbreviation	Char(16)	N	An Abbreviation of the audio type of a feature (a film).
Description	Varchar(30)	Y	Description of the audio type.

### Indexes

Name	Kind	Columns	Purpose
PKRMAudioTypesAudioTypeID	P	AudioTypeID	Primary Key.

## 18.9 RMCapacity

The RMCapacity table contains capacity records for all events and capacity managed resources.

### Columns

Column	Type	Allow Nulls	Description
CapacityID	Int	N	Primary key, always unique
EventID	Int	N	Foreign key to RMEvents.EventID. This is the event that this capacity record is associated with.
Available	Int	N	Capacity available for the associated event.
Sold	Int	N	Capacity sold for the associated event.
QuantityOnHold	Int	N	Capacity amount being used in an ongoing transaction, stored in the CapacityHold table.
ResourceID	Int	N	Foreign key to RMResources.ResourceID. This is the resource that this capacity record is associated with.
TotalCapacity	Int	N	Represents the Total Capacity as defined in the Resource's Maximum Capacity field (RMResources.MaxAllowed). This field is initialized to this value upon event creation.
Sequence	Int	N	This zero-based number indicates the order of removing capacity from a multiple capacity-managed resource.
Name	VarChar(50)	Yes	The name of a capacity hold
Description	VarChar(255)	Yes	The description of a capacity hold
CapacityType	Integer	No	The type of capacity <sup>1</sup>
ReleaseState	Integer	No	The status of a hold capacity <sup>2</sup>
ReleasedCapacity	Integer	Yes	Quantity of capacity that was released when hold capacity ReleaseState became WAS_RELEASED
HoldGroupID	Integer	Yes	GatewayCounter used to group event hold capacity records together

### Indexes

Name	Kind	Columns	Purpose
PKRMCapacityCapacityID	P	CapacityID	Primary key, always unique.
IXRMCapacityEventIDResourceID		EventID, ResourceID	Used for event picklists, looking up capacity for resources.
IXCapacityTypEventIDCapacityID		CapacityType, EventID, CapacityID	To improve query performance when searching by capacity type.

### <sup>1</sup> CapacityType Values

Value	Gateway Constant Name	Description
0	GENERAL_CAPACITY	Resource capacity record
1	HOLD_CAPACITY	Resource capacity being held back from general sales

### <sup>2</sup> ReleaseState Values

Value	Gateway Constant Name	Description
0	NOT_RELEASEABLE	Indicates that a hold capacity may NOT be released for general sales
1	IS_RELEASEABLE	Indicates that a hold capacity may be released for general sales
2	WAS_RELEASED	Indicates that a hold capacity was released for general sales

## 18.10 RMCapacityHold

The RMCapacityHold table contains temporary capacity information to allow a node to recover capacity if the machine goes down in the middle of a sale. When Galaxy starts up, this table is checked for the current node number. If any entries are found for the current node, the capacity is replenished for that event, and the record is removed from this table.

### Columns

Column	Type	Allow Nulls	Description
UniqueHoldID	Int	N	Primary key, always unique.
EventID	Int	N	Foreign key to RMEvents.EventID. This is the event that this capacity hold record is associated with.
Node	Int	N	Node number of the machine holding the capacity.
HoldQuantity	Int	N	Quantity that is being held.
ResourceID	Int	N	Foreign key to RMResources.ResourceID. This is the resource that this capacity hold record is associated with.
DateTimeHeld	DateTime	N	Date and time this quantity was held - for tracking purposes.
GroupIDType	Integer	Y	Identifies the contents of the GroupID column. <sup>1</sup>
GroupID	Integer	Y	A numeric code uniquely identifying the purpose of this record.
CapacityID	Integer	Yes	The unique ID of the related RMCapacity record

### Indexes

Name	Kind	Columns	Purpose
PKRMCapacityHoldUniqueHoldID	P	UniqueHoldID	Primary key, always unique.
IXRMCapacityHoldEventIDNodeRID		EventID, Node, ResourceID	Used for checking table at startup.
IXRMCapacityHoldGrpIDTypeGrpID		GroupIDType, GroupID	Used to link a Capacity Hold record to a particular Order.
IXRMCptyHldEvtIDRrcIDGpTIDGrID		EventID, ResourceID, GroupIDType, GroupID, CapacityID	Improve performance
IXAdjustCapacityHoldDelete		EventID, ResourceID, GroupIDType, GroupID, CapacityID, Node, HoldQuantity	Necessary when deleting from GTS_SP_RM_AdjustCapacityHold

<sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	GroupID Meaning
0	CAPGRP_POS	Zero or NULL. Capacity record for POS. The value of <i>GroupID</i> is zero.
1	CAPGRP_ORDER	<i>GroupID</i> is a Foreign key to Orders.OrderID.
2	CAPGRP_EGALAXY	Current record is for eGalaxy

## 18.11 RMCommissions

Store the commission rate associated to an event type.

### Columns

Column	Type	Allow Nulls	Description
RMCommissionID	Int	N	Primary key, always unique.
Name	Char(20)	N	Name of the commission rate
EventType	Int	Y	Event Type ID
CommissionType	Int	Y	Commission Type ID
Basis	Int	N	P or F - Percentage or Flat amount
Amount	Float	N	Either the percentage or flat amount as per the basis field

### Indexes

Name	Kind	Columns	Purpose
PKRMCommissionsRMCommissionID	P	RMCommissionID	Primary Key.
IXRMCommissionsName		Name	Name sort for picklist displays

### <sup>1</sup> GroupIDType Values

Value	Gateway Constant Name	Description
0	BASIS_PERCENT	The <i>Amount</i> is a percentage
1	BASIS_FLATRATE	The <i>Amount</i> is a flat rate

## 18.12 RMCommissionTypes

This table is a way to categorize the Commissions in the RMCommissions Table.

### Columns

Column	Type	Allow Nulls	Description
RMCommissionTypeID	Int	N	Primary key, always unique.
Name	Char(20)	N	Name of the commission type
Description	Varchar(30)	Y	Description of the type

### Indexes

Name	Kind	Columns	Purpose
PKCommissionTypesRMCommTypeID	P	RMCommissionTypeID	Primary Key.
IXCommissionTypesName		Name	Name order for picklists

### 18.13 RMDistributors

The RMDistributors table contains groups who are responsible for releasing a film. Distributor information is used in cinema ticketing to report commission for a feature.

#### Columns

Column	Type	Allow Nulls	Description
DistributorID	Int	N	Primary key, always unique.
DistributorCode	Char(16)	N	Code assigned to the distributor. Must contain a unique value for each Distributor.
Name	Varchar(50)	N	Name of the distributor.
Street1	Varchar(36)	Y	Street address, first line.
Street2	Varchar(36)	Y	Street address, second line.
Street3	Varchar(36)	Y	Street address, third line.
City	Varchar(40)	Y	City.
State	Varchar(40)	Y	State / Province.
Postal	Char(16)	Y	Postal code (ZIP code for USA).
CountryCode	Char(2)	Y	Country Code.

#### Indexes

Name	Kind	Columns	Purpose
PKRMDistributorsDistributorID	P	DistributorID	Primary Key.
AKRMDistributorsDistCode	A	DistributorCode	Alternate Key.

## 18.14 RMEventPrograms

The RMEventPrograms table contains all of the template Event information to be used when creating new Events

### Columns

Column	Type	Allow Nulls	Description
EventProgramID	Int	N	Primary key, always unique
EventProgramName	VarChar(100)	N	Name of the event program
PreventDynamicEventUse	Bit	Y	If enabled (1), cannot be used to when creating a dynamic event
OverrideCapacity	Bit	Y	If enabled (1), the capacity of the event will be overridden with the value of the CapacityValue field.
CapacityValue	Int	Y	If overriding capacity on the event, the value of this field will be used for the capacity of the event
EventTypeID	Int	N	Foreign key to RMEventTypes.EventTypeID. The event type associated to this program must be specified.
DurationInMinutes	Int	Y	Default duration for the event. The GUI splits this up between hours and minutes.
ReserveExclusive	Bit	N	Determines whether the reservation made for the event associated with this event program are made exclusively or not.

### Indexes

Name	Kind	Columns	Purpose
PKEventProgramID	P	EventProgramID	Primary Key.
IXRMEventProgramsEventTypeID	IX	EventTypeID	Load the list of Event Programs via Event Type

## 18.15 RMEventProgramDetails

The RMEventProgramDetails table contains details associated to entries in the RMEventPrograms table.

### Columns

Column	Type	Allow Nulls	Description
EventProgramDetailID	Int	N	Primary key, always unique
EventProgramID	Int	N	Foreign key to RMEventPrograms table
DetailType	Int	N	Defines the type of this detail record <sup>1</sup>
DetailID	Int	N	Foreign key to RMResourceTypes table or RMResources table, depending on the value of the DetailType column
PrimaryDetail	Bit	Y	Defines whether this Event Program Detail is the Event Programs Primary Event Program Detail
RequiredQty	Int	N	Number of required resources or resource types that must be selected when creating an event
OptionalQty	Int	N	Number of optional resources or resource types that can be selected when creating an event
Sequence	Int	N	Defines the sequence of how the details are processed within the event wizard.

### Indexes

Name	Kind	Columns	Purpose
PKEventProgramDetailID	P	EventProgramDetailID	Primary Key.
IXEventProgDtlsEventProgramID		EventProgramID	Load the detail list per Event Program

<sup>1</sup> DetailType Values

Value	Gateway Constant Name	Description
0	EVENT_PROGRAM_RESOURCE_TYPE_DETAIL	DetailID references RMResourceTypes table
1	EVENT_PROGRAM_RESOURCE_DETAIL	DetailID references RMResources table

## 18.16 RMEvents

The **RMEvents** table contains all events defined in the system. An event has foreign keys to several tables, including RMEventTypes, RMResources and RMReservations.

Columns

Column	Type	Allow Nulls	Description
EventID	Int	N	Primary key, always unique.
EventName	Char(25)	N	Name of the event.
StartTime	DateTime	N	Start date and start time of the event.
EndTime	DateTime	N	End date and end time of the event.
NotesID	Int	N	Foreign key to Notes.NoteID. Represents the note record for this event. If this value is zero, there are no defined notes. A value greater than zero needs to point to a valid record in the Notes table.
EventTypeID	Int	N	Foreign key to RMEventTypes.EventTypeID. Represents the event type for this event. This value must be greater than zero and point to a valid record in the RMEventTypes table.
OnSaleDateTime	DateTime	N	Date and Time when you can start selling tickets for this event.
OffSaleDateTime	DateTime	N	Date and Time when this event stops being sellable.
ResourceID	Int	N	Foreign key to RMResources.ResourceID. Represents the resource for this event. This value must be greater than zero and point to a valid record in the RMResources table.
ResourceReservationID	Int	N	Foreign key to RMReservations.ReservationID. Represents the reservation for this event. This value must be greater than zero and point to a valid record in the RMReservations table.
UserEventNumber	Int	Y	A user-defined number to be associated with each event. It is editable and can be auto-generated using the Auto Create Events function.
ActiveIndicator	Int	Y	Code representing whether or not the event is active. <sup>1</sup>
LimitIndicator	Int	Y	This field will eventually be used to determine whether or not this event can be oversold. <i>This column is not currently used by the system.</i>
ShowID	Int	Y	Foreign key to RMShows.ShowID, representing the show assigned for this event. It can be NULL as some events might not require a show.
PrivateEvent	Bit	Y	If enabled (1), can only sell for that event if a) No tickets have been sold for the event, or b) You have tickets for that event already on an existing order.
EventProgramID	Int	Y	Foreign Key to RMEventPrograms table. Denotes that an Event Program was used to create the Event.
NodeNumber	Int	Y	If this event was created on-the-fly during Event Availability and is marked as Private, the machine's node number will be used for this column. This prevents other nodes from stealing the event. This field is NOT editable in the GUI. It can be cleared from the Edit Event Screen, but not edited.
AttributeValueGroupID	Int	Y	Points to the Attribute Values Group
RosterAttributeGroupID	Int	Y	Reference to group of associated attributes
HasRoster	Bit	Y	Determines if the Event will be collecting roster information
HasWaitList	Bit	Y	Determines if the event can use a wait list.
WaitListCodeTableID	Int	Y	The foreign key to the CodeTableID on CodeTables
RSEventSeatMapID	Int	Y	FK link to RSEventSeatMaps
HasSequencedCapacity	Bit	Y	Denotes whether or not the event is managing more than one capacity-managed resource. It is true when there are more than one row in the RMCapacity table associated to this event with non-zero RMCapacity.Sequence values.
HasEventSaleLimits	Bit	Y	Denotes whether or not this event has any Event Sale Limit information associated with it.
EnableSeatOptions	Bit	Y	Determines if the Event Management Service will return multiple seat options for best available requests for this event.
SeatOptionsToReturn	Int	Y	The number of seat options that will be returned for this event.
SeatOptionsPercentage	Int	Y	The percentage of seats that must be available to reserve for seat options to be sent.
MaxSeatsPerSession	Int	Y	The maximum number of seats a session can reserve for this event.
EnableSingleSeatCheck	Bit	Y	If true prevents single seats to be left in a row when calling best available.
EventGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

Indexes

Name	Kind	Columns	Purpose
PKRMEventsEventID	P	EventID	Primary key, always unique.
IXRMEventsStartTime		StartTime	Used for filtering, sorting picklist results.
IXRMEventsEventTypeID		EventTypeID	Used for filtering results.
IXRMEventsEndTimeOnSaleOffSale		EndTime, OnSaleDateTime, OffSaleDateTime	Used for querying for "first available" event. (Index name does not conform to standards due to 30 character name limit in Oracle).
IXRMEventsEventIDRosterGroupID		EventID, RosterAttributeGroupID	Used for queries based on EventID and AttributeGroupID and RosterAttributeGroupID
IXRMEventsActIndStartEvtDevTyp		ActiveIndicator, StartDatetime, EventID, EventTypeID	For queries based on ActiveIndicator, startDateTime.

<sup>1</sup> ActiveIndicator Values

Value	Description
0	Event is Active
1	Event is Inactive

## 18.17 RMEventTypes

The RMEventTypes table consists of types of events, to categorize events in the RMEvents table.

### Columns

Column	Type	Allow Nulls	Description
EventTypeID	Int	N	Primary key, always unique.
Name	VarChar(40)	N	The name of the event type.
AttributeGroupID	Integer	Y	Points to the Attribute Values Group
RosterAttributeGroupID	Int	Y	Reference to group of associated attributes
HasRoster	Bit	Y	Determines if the event type supports collecting roster information
HasWaitList	Bit	Y	Determines if the event can use a wait list.
WaitListCodeTableID	Int	Y	The foreign key to the CodeTableID on CodeTables
CalendarDisplay	Int	Y	A number indicating the status of the calendar display on the webstore <sup>1</sup> .
EventTypeGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKRMEventTypesEventTypeID	P	EventTypeID	Primary key, always unique.
IXRMEventTypeName		Name	Used for picklist.

<sup>1</sup> CalendarDisplay Values

Value	Gateway Constant Name	Description
0	EVENT_TYPE_CALENDAR_DISPLAY_NONE	No selection - defaults to the calendar options defined on the Sales Channel Category
1	EVENT_TYPE_CALENDAR_DISPLAY_DATE_TIME	Display the date selection first and then the available event times.
2	EVENT_TYPE_CALENDAR_DISPLAY_DATE	Only display the date selection. Do not display the time selection.
3	EVENT_TYPE_CALENDAR_DISPLAY_EVENT	No calendar to be displayed. No date and time picker. The event is the first thing displayed.

## 18.18 RMFeatures

The RMFeatures table contains the definition of a feature such as the principal motion picture, exhibition, or performance in a program. For cinemas, a feature represents a film and is used to track ticket sales by that film for reporting to distributors. The attributes of a feature typically include Duration (in minutes), Rating (i.e. PG, PG-13, R) and Distributor.

### Columns

Column	Type	Allow Nulls	Description
FeatureID	Int	N	Primary key, always unique.
FeatureCode	Char(16)	N	Code assigned to the feature. Must contain a unique value for each feature.
Name	Varchar(50)	N	Name of the feature.
ShortName	Char(16)	Y	Short name of the feature.
DistributorID	Int	N	Foreign key to RMDistributors.DistributorID, representing the distributor of this feature. Must contain a value greater than zero, and point to a valid distributor.
FeatureRatingID	Int	N	Foreign key to RMFeatureRatings.FeatureRatingID, representing the rating assigned for this feature. Must contain a value greater than zero, and point to a valid feature rating.
RatingExplanation	Varchar(50)	Y	Brief explanation of why a feature (or a film) has received its rating.
Duration	Int	N	Duration of a feature in minutes. A feature must have a duration.
ReleaseNumber	Char(16)	Y	Internal release number assigned to each feature by a theater.
FeatureTypeID	Int	Y	Foreign key to RMFeatureTypes.FeatureTypeID, representing the type of this feature. Must contain a value greater than zero, and point to a valid feature type.
AudioTypeID	Int	Y	Foreign key to RMAudioTypes.AudioTypeID, representing the format of the audio used by this feature. Must contain a value greater than zero, and point to a valid audio type.
Notes	Text	Y	Contains a brief synopsis of the feature or a film.

### Indexes

Name	Kind	Columns	Purpose
PKFeatureID	P	FeatureID	Primary Key.
AKRMFeaturesFeatureCode	A	FeatureCode	Alternate Key
IXRMFeaturesName		Name	Used by the query that is executed from the feature picklist
IXRMFeaturesDistributorID		DistributorID	Used by the query that is executed from the feature picklist
IXRMFeaturesFeatureRatingID		FeatureRatingID	Used by the query that is executed from the feature picklist
IXRMFeaturesReleaseNumber		ReleaseNumber	Used by the query that is executed from the feature picklist

## 18.19 RMFeatureTypes

The **RMFeatureTypes** table contains types of a film. A film can be a comedy or an action or a science fiction. These are all different types of a feature and are assigned to a feature when defining one.

### Columns

Column	Type	Allow Nulls	Description
FeatureTypeID	Int	N	Primary key, always unique.
Abbreviation	Char(8)	N	An abbreviation of the feature type.
Name	Varchar(30)	N	Name of the feature type.

### Indexes

Name	Kind	Columns	Purpose
PKRMFeatureTypesFeatureTypeID	P	FeatureTypeID	Primary Key.

## 18.20 RMFeatureRatings

The **RMFeatureRatings** table contains the ratings that are assigned to a film. The rating system is a guide that helps you determine the movies content prior to viewing it. The feature rating is used when defining a feature.

### Columns

Column	Type	Allow Nulls	Description
FeatureRatingID	Int	N	Primary key, always unique.
Abbreviation	Char(8)	N	An Abbreviation of the rating of a feature (a film).
Name	Varchar(30)	Y	Name of a feature rating.
Description	Varchar(50)	Y	Brief description of a feature rating.

### Indexes

Name	Kind	Columns	Purpose
PKRMFeatRatingsFeatureRatingID	P	FeatureRatingID	Primary Key.

## 18.21 RMReservationHold

The RMReservationHold table contains temporary reservation information to allow a node to recover resource reservations if the machine goes down in the middle of a sale. When Galaxy starts up, this table is checked for the current node number. If any entries are found for the current node, the reservations for those resources are cancelled, and the record is removed from this table.

Columns

Column	Type	Allow Nulls	Description
ReservationHoldID	Int	N	Primary key, always unique.
ReservationID	Int	N	Foreign key to RMResources.ResourceID. Represents the resource for this reservation hold. This value must be greater than zero and point to a valid record in the RMResources table.
Node	Int	N	Node number of the machine holding the Reservation.
DateTimeHeld	DateTime	N	Date and time the Reservation was held - for tracking purposes.

Indexes

Name	Kind	Columns	Purpose
PKReservationHoldID	P	ReservationID	Primary key, always unique (Index name does not conform to standards due to 30 character name limit in Oracle).
IXRMReservationHoldNode		Node	Used for removing records by node.
IXRMReservationHoldReserveID		ReservationID	Used for removing records by node and ReservationID. (Index name does not conform to standards due to 30 character name limit in Oracle).

## 18.22 RMReservations

The RMReservations table contains reservations for events, capacity managed resources and plain resource reservations.

Columns

Column	Type	Allow Nulls	Description
ReservationID	Int	N	Primary key, always unique.
StartTime	DateTime	N	Start date and start time of the reservation.
EndTime	DateTime	N	End date and end time of the reservation.
ResourceID	Int	N	Foreign key to RMResources.ResourceID. Represents the resource that is reserved. This value must be greater than zero and point to a valid record in the RMResources table.
ResourceTypeID	Int	N	Foreign key to RMResourceTypes.ResourceTypeID. Represents the type of resource that is reserved. This value must be greater than zero and point to a valid record in the RMResourceTypes table.
NotesID	Int	N	Foreign key to Notes.NoteID. Represents the note record for this reservation. If this value is zero, there are no defined notes. A value greater than zero needs to point to a valid record in the Notes table.
ReserveExclusive	Bit	N	Determines if the reservation is exclusive or not. If the reservation is exclusive, no other reservation can be made during that time frame for the Resource specified by ResourceID.
EventID	Int	Y	Foreign key to RMEvents.EventID, used to associate multiple resources to an event.

Indexes

Name	Kind	Columns	Purpose
PKRMReservationsReservationID	P	ReservationID	Primary key, always unique.
IXRMReservationsStartTime		StartTime, EndTime, ResourceID, ResourceTypeID	Used for sorting picklists and for all queries. (Index name does not conform to standards due to 30 character name limit in Oracle).
IXRMReservationsEventID		EventID	Used when querying for all reservations for a given event

### 18.23 RMResources

The RMResources table contains all of the defined Resources in the system. A Resource can be a person, a room, a picnic area, or pretty much anything that can be reserved.

#### Columns

Column	Type	Allow Nulls	Description
ResourceID	Int	N	Primary key, always unique
AncestorID	Int	N	Resource ID of the parent of this resource. If the value is 0, then this resource is a parent.
ResourceName	Char(50)	N	Name of the resource.
ResourceTypeID	Int	N	Foreign key to RMResourceTypes.ResourceTypeID. Represents the type of resource. This value must be greater than zero and point to a valid record in the RMResourceTypes table.
ManageCapacity	Bit	N	Determines if capacity is managed for this resource.
MinNeeded	Int	Y	Minimum number of people needed for this resource.
MaxAllowed	Int	Y	Maximum number of people needed for this resource (Maximum Capacity).
ManageChildCapacity	Bit	N	Determines if "child capacity" is managed for this resource. If child capacity is managed, then the combined child resource capacities needs to be less than or equal to the maximum capacity on the parent resource.
AttributeValueGroupID	Integer	Y	Points to the Attribute Values Group
ResourceGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKRMResourcesResourceID	P	ResourceID	Primary key, always unique.
IXRMResourcesNameTypeID		ResourceName, ResourceTypeID	For sorting, querying picklists. (Index name does not conform to standards due to 30 character name limit in Oracle).

## 18.24 RMResourceTypes

The RMResourceTypes table contains types of resources, to categorize resources in the RMResources table. These records are also used as "templates" for Resources, for default values when defining resources.

### Columns

Column	Type	Allow Nulls	Description
ResourceTypeID	Int	N	Primary key, always unique
Name	Char(25)	N	Name of the Resource type
ManageCapacity	Bit	N	Determines if capacity is managed for this resource
MinNeeded	Int	Y	Minimum number of people needed for this resource type
MaxAllowed	Int	Y	Maximum number of people needed for this resource type (Maximum Capacity)
ManageChildCapacity	Bit	N	Determines if Child Capacity is managed for this Resource Type. If child capacity is managed, then the combined child capacity needs to be less than or equal to the maximum capacity on the parent resource
AttributeGroupID	Integer	Y	Points to the Attribute Values Group

### Indexes

Name	Kind	Columns	Purpose
PKRMResourceTypesResTypeID	P	ResourceTypeID	Primary key, always unique. (Index name does not conform to standards due to 30 character name limit in Oracle).
IXRMResourceTypesName		ResourceTypeName	Used for sorting picklist. (Index name does not conform to standards due to 30 character name limit in Oracle).

## 18.25 RMShows

The RMShows table contains all shows defined in the system. A show is a program with a feature and associated details including lead up, advertisement, trailer, feature duration, and cleanup time.

### Columns

Column	Type	Allow Nulls	Description
ShowID	Int	N	Primary key, always unique.
ShowCode	Char(16)	N	Code assigned to the show. Must contain a unique value for each show.
Name	Varchar(50)	N	Name of the show.
ShortName	Char(16)	Y	Short name of the show.
FeatureID	Int	N	Foreign key to RMFeatures.FeatureID, representing the feature of this show. Must contain a value greater than zero, and point to a valid feature.
LeadUpInterval	Int	Y	Lead up time in minutes.
AdvertisementInterval	Int	Y	Advertisement time in minutes.
TrailerInterval	Int	Y	Trailer time in minutes.
CleanUpInterval	Int	Y	CleanUp time in minutes.
RunFromDate	DateTime	Y	Date and time of the first run of this show.
RunThruDate	DateTime	Y	Date and time of the last run of this show.

### Indexes

Name	Kind	Columns	Purpose
PKRMShowsShowID	P	ShowID	Primary Key.
AKRMShowsShowCode	A	ShowCode	Alternate Key
IXRMShowsName		Name	Used by the query that is executed from the show picklist
IXRMShowsFeatureID		FeatureID	Used by the query that is executed from the show picklist
IXRMShowsRunFromRunThruDate		RunFromDate, RunThruDate	Used by the query that is executed from the show picklist

## 18.26 RSEventPrices

The RSEventPrices table provides the Price Plan for an actual reserved seating event.

### Columns

Column	Type	Allow Nulls	Description
RSEventPriceID	Identity	N	Primary key, always unique
RSEventSeatMapID	Integer	N	FK to RSEventSeatMap
Name	Varchar(128)	N	Name of the Price Plan
Color	Integer	Y	Color identifier that will be used when drawing the seat map with pricing plans
Rank	Integer	Y	Used to determine the next price level when looking at the best available seat options.

### Indexes

Name	Kind	Columns	Purpose
PKRSEventPriceID	P	RSEventPriceID	Primary Key
IXRSEventPricesSeatMapID		RSEventSeatMapID	Speed query performance

## 18.27 RSEventPriceDetails

The RSEventPriceDetails table provides the details of the Price Plan for an actual reserved seating event. This detail table identifies the product PLU codes that are included with a Price Plan for an actual reserved seating event. Currently only supports one PLU per reserved seating event Price Plan.

### Columns

Column	Type	Allow Nulls	Description
RSEventPriceDetailID	Identity	N	Primary key, always unique
RSEventPriceID	Int	N	FK to RSEventPrices
PLU	Varchar(20)	N	The PLU code of the product that will be included as part of the associated price plan for the actual reserved seating event.
SalesChannelID	Integer	Y	The SalesChannelID of the sales channel that will be included as part of the associated price plan for the actual reserved seating event.

### Indexes

Name	Kind	Columns	Purpose
PKRSEventPriceDetailID	P	RSEventPriceDetailID	Primary Key

## 18.28 RSEventSeatHolds

The RSEventSeatHolds table defines the hold code that is associate to a live event seat. This table controls the seat hold association at the live event level. The associate for the template is stored in RSSeatHolds.

### Columns

Column	Type	Allow Nulls	Description
RSEventSeatHoldID	Identity	N	Primary key, always unique
RSEventSeatID	Integer	N	FK link to RSEventSeats.RSEventSeatID. This is the unique Seat ID from the live event reserved seating map.
RSHoldCodeID	Integer	N	FK link to RSHoldCodes.RSHoldCodeID. This is the unique hold code ID from the Hold Codes table.

### Indexes

Name	Kind	Columns	Purpose
PKRSEventSeatHoldID	P	RSEventSeatHoldID	Primary Key
IXRSEventSeatHoldsSeatID		RSEventSeatID	Speed query performance

## 18.29 RSEventSeatMapObjects

The RSEventSeatMapObjects table contains a record for non-seat objects created for an actual reserved seating event.

### Columns

Column	Type	Allow Nulls	Description
RSEventSeatMapObjectID	Identity	N	Primary key, always unique.
RSEventSeatMapID	Integer	N	FK to the RSEventSeatMaps table
ObjectKind	Integer	N	Type of shape <sup>1</sup>
CoordX	Integer	N	Left coordinate for placement on the seat map
CoordY	Integer	N	Top coordinate for placement on the seat map
Width	Integer	N	Value of the width of the shape
Height	Integer	N	Value of the height of the shape
Description	nvarchar(10)	Y	A brief description to be displayed on the seat map to note what the object represents
Color	Integer	Y	The Color of the seat map object
TextColor	Integer	Y	The Color of the description text

### Indexes

Name	Kind	Columns	Purpose
PKRSEventSeatMapObjectID	PK	RSEventSeatMapObjectID	Table primary key
IXRSEventPricesSeatMapObjSeatMapID		RSEventSeatMapID	Speed query performance

<sup>1</sup> ObjectKind Values

Value	Gateway Constant Name	Description
0	okRectangle	Rectangle object kind
1	okEllipse	Ellipse object kind

### 18.30 RSEventSeatMaps

The **RSEventSeatMaps** table provides a link between an event and the reserved seating map selected for the event.

#### Columns

Column	Type	Allow Nulls	Description
RSEventSeatMapID	Identity	N	Primary key, always unique.
RSSeatMapID	Integer	Y	FK link to RSSeatMaps table
EventID	Integer	N	Fk Link to RMEvents Table
Available	Integer	Y	The number of seats that are available to be reserved
Reserved	Integer	Y	The number of seats that are reserved
Sold	Integer	Y	The number of seats that are sold

#### Indexes

Name	Kind	Columns	Purpose
PKRSEvtSeatMapsRSEvtSeatMapID	P	RSEventSeatMapID	Primary Key.

### 18.31 RSEventSeats

The RSEventSeats table contains a record for each seat created for an actual reserved seating event.

#### Columns

Column	Type	Allow Nulls	Description
RSEventSeatID	Identity	N	Primary key, always unique.
RSEventSeatMapID	Integer	N	Link to RSEventSeatMaps table
RSSeatID	Integer	Y	Link to RSSeats (can be null if seat does not exist in seat map)
SectionName	Char(10)	N	Name of the section in which the seat exists
RowName	Char(10)	N	Name of the row in which the seat exists
SeatName	Char(10)	N	Name that distinguishes the seat within the row
SeatStatus	Integer	N	Current seat status <sup>1</sup>
VisualID	VarChar(40)	Y	VisualID of ticket for the seat (if any)…
CoordX	Integer	N	Used in positioning seat in graphic seat map
CoordY	Integer	N	Used in positioning seat in graphic seat map
Rank	Integer	N	Used in determining 'best available'
Adjacency	Integer	N	Used to determine adjacent seats
OrderID	Integer	Y	Link to Orders table
NodeNo	Integer	Y	
DateTimeHeld	DateTime	Y	Used to prevent indefinite holds on seats
SessionID	Integer	Y	Web session id
TransactionID	Integer	Y	ID of database transaction updating this table
RSPPriceID	Integer	Y	FK to RSPPrices.RSPPriceID
RSEventPriceID	Integer	Y	FK to RSEventPrices
SessionGuid	UniqueIdentifier	Y	A unique identifier used to designate who has reserved the seat.
ContactID	Int	Y	FK reference to the CustContacts table for seats purchased with PLUs that require a contact.

#### Indexes

Name	Kind	Columns	Purpose
PKRSEventSeatsRSEventSeatID	P	RSEventSeatID	Primary Key.
IXRSEventSeatsVisualID		VisualID	Speed query performance
IXRSEventSeatsSeatMapID		RSEventSeatMapID	Speed query performance

#### <sup>1</sup> SeatStatus Values

Value	Gateway Constant Name	Description
0		Available
1		Reserved - seat is only available to the transaction or order that changed the seat to the "Reserved" status.
2		Sold
3		Hold - Not on sale

### 18.32 RSEventSections

The RSEventSections table contains a record for each section in an actual reserved seating event.

#### Columns

Column	Type	Allow Nulls	Description
RSEventSectionID	Identity	N	Primary key, always unique.
RSEventSeatMapID	Integer	N	FK to the RSEventSeatMap table
RSSectionID	Integer	N	FK to the RSSection table that was the template for this section
SectionName	nvarchar(10)	N	The name of the section to help distinguish it in the seat map
SeatOrientation	Integer	N	Indicates the numbers scheme for rows and seats: LF, RF, etc.
SeatCount	Integer	N	The number of seats in this section
Rank	Integer	N	Used to determine the next section when using the best available seat options
PictureID	Integer	Y	FK to a record the Pictures table that holds an image displaying a view of the stage from the section.

#### Indexes

Name	Kind	Columns	Purpose
PKRSEventSectionID	P	RSEventSectionID	Primary Key.

### 18.33 RSHoldCodes

The RSHoldCodes table defines hold codes for reserve seating. A hold code can be associated with a seat and can be privilege based.

#### Columns

Column	Type	Allow Nulls	Description
RSHoldCodeID	Identity	N	Primary key, always unique
Code	Char(20)	N	A two-letter code that can be used to describe the hold. This identifier is used on seat maps for visual representation of the seat hold.
Name	varchar(80)	Y	The name character value that is set by the user during the configuration process.
Description	Text	Y	The description character value that is set by the user during the configuration process.

#### Indexes

Name	Kind	Columns	Purpose
PKRSHoldCodeID	P	RSHoldCodeID	Primary Key

### 18.34 RSHoldCodeUserProfiles

The RSHoldCodeUserProfiles table defines the user profiles that have the privilege to select or release a seat from a hold status.

#### Columns

Column	Type	Allow Nulls	Description
RSHoldCodeUserProfileID	Identity	N	Primary key, always unique
RSHoldCodeID	Integer	N	FK link to RSHoldCodes.RSHoldCodeID. This is the ID of the hold code that will be linked to a user profile.
ClassID	Char(10)	N	FK link to UserProfiles.Name. This is the name of the User Profile that will have privilege to this hold code.

#### Indexes

Name	Kind	Columns	Purpose
PKRSHoldCodeUserProfileID	P	RSHoldCodeUserProfileID	Primary Key

### 18.35 RSPrices

The RSPrices table provides the Price Plan for a reserved seating map. A group of seats, rows, or sections can be identified as a particular price plan, which then determines the PLU that will be used for that price plan.

#### Columns

Column	Type	Allow Nulls	Description
RSPriceID	Identity	N	Primary key, always unique
Name	varchar(128)	N	Name of the Price Plan
Color	Integer	Y	Color identifier that will be used when drawing the seat map with price plans

#### Indexes

Name	Kind	Columns	Purpose
PKRSPriceID	P	RSPriceID	Primary Key

### 18.36 RSPriceDetails

The RSPriceDetails table provides the details of the Price Plan for a reserved seating map. This detail table identifies the product PLU codes that are included with a Price Plan. In the first iteration of Reserved Seating, the Price Plan will support one product PLU.

#### Columns

Column	Type	Allow Nulls	Description
RSPriceDetailID	Identity	N	Primary key, always unique
PLU	varchar(20)	N	The PLU code of the product that will be included as part of the associated price plan.
RSPricelD	Integer	Y	FK link to RSPrices.RSPricelD. This is the ID of the Price Plan that this detail record links to.
SalesChannelID	Integer	Y	The Sales Channel code of the Sales Channel that will be included as part of the associated price plan.

#### Indexes

Name	Kind	Columns	Purpose
PKRSPriceDetailID	P	RSPriceDetailID	Primary Key

### 18.37 RSResourceMaps

The **RSResourceMaps** is a "connections" table supporting the many-to-many relationship between Resources and RSSeatMaps.

RSResourceMaps relates a resource to any seat maps that have been defined for the resource. Multiple seating configurations for a resource may be defined this way. More than one resource may share a seat map and resources that don't support reserved seating will not have any seat maps.

#### Columns

Column	Type	Allow Nulls	Description
RSResourceMapID	Identity	N	Primary key, always unique.
ResourceID	Integer	N	FK link to RMResource table
RSSeatMapID	Integer	N	FK link to RSSeatMaps Table

#### Indexes

Name	Kind	Columns	Purpose
PKRSResrcMapsRSResourceMapID	P	RSResourceMapID	Primary Key.

### 18.38 RSRowPrices

The RSRowPrices table can be used to relate a price plan with a row of a reserved seating map. The price plan will be set at this level if it is not overridden by configuration at the seat level; the relative table is RSSeatPrices. The parent tables to this are RSSectionPrices and RSSeatMapPrices.

#### Columns

Column	Type	Allow Nulls	Description
RSRowPriceID	Identity	N	Primary key, always unique
RSSeatMapPriceID	Integer	N	FK link to RSSeatMapPrices.RSSeatMapPriceID. This is the ID of the reserved seating map price record, the master parent to this record.
SectionName	Char(10)	N	The name character value that is set by the user during the configuration process.
RowName	Char(10)	N	The name character value that is set by the user during the configuration process.
RSPricelD	Integer	Y	FK link to RSPrices.RSPricelD. This is the ID of the Price Plan that will be used for this reserved seating row.
SectionID	Integer	N	FK to the RSSections table
RowID	Integer	N	FK to RSRows table

#### Indexes

Name	Kind	Columns	Purpose
PKRSRowPriceID	P	RSRowPriceID	Primary Key.

### 18.39 RSRows

The **RSRows** table contains a record for every row created for a seat map. The same row name may appear in more than one section.

#### Columns

Column	Type	Allow Nulls	Description
RSRowID	Identity	N	Primary key, always unique.
RSSectionID	Integer	N	FK link to RSSections
RowName	Char(10)	N	Name that identifies the row within the section

#### Indexes

Name	Kind	Columns	Purpose
PKRSRowsRSRowID	P	RSRowID	Primary Key.

## 18.40 RSSeatHolds

The RSSeatHolds table defines the hold code that is associated to a template seat. This table controls the seat hold association at a template level. The actual association to a live reserved seating event is stored in RSEventSeatHolds.

### Columns

Column	Type	Allow Nulls	Description
RSSeatHoldID	Identity	N	Primary key, always unique
RSSeatID	Integer	N	FK link to RSSeats.RSSeatID. This is the unique Seat ID from the template configuration of the reserved seating map.
RSHoldCodeID	Integer	N	FK link to RSHoldCodes.RSHoldCodeID. This is the unique hold code ID from the Hold Codes table.

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatHoldID	P	RSSeatHoldID	Primary Key

## 18.41 RSSeatMapObjects

The RSSeatMapObjects table contain objects used to denote structures on the seat map other than the seats themselves.

### Columns

Column	Type	Allow Nulls	Description
RSSeatMapObjectID	Identity	N	Primary key, always unique.
RSSeatMapID	Integer	N	FK to the RSSeatMaps table
ObjectKind	Integer	N	Type of shape <sup>1</sup>
CoordX	Integer	N	Left coordinate for placement on the seat map
CoordY	Integer	N	Top coordinate for placement on the seat map
Width	Integer	N	Value of the width of the shape
Height	Integer	N	Value of the height of the shape
Description	nvarchar(10)	Y	A brief description to be displayed on the seat map to note what the object represents
Color	Integer	Y	The Color of the seat map object
TextColor	Integer	Y	The Color of the description text

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatMapObjectID	PK	RSSeatMapObjectID	Table primary key

### <sup>1</sup> ObjectKind Values

Value	Gateway Constant Name	Description
0	okRectangle	Rectangle object kind
1	okEllipse	Ellipse object kind

## 18.42 RSSeatMapPriceConnections

The RSSeatMapPriceConnections table is used to link a Seat Map to a Seat Map Price Configuration. A seat map

could have multiple price configurations. This table connects a seat map to one or many price configurations.

### Columns

Column	Type	Allow Nulls	Description
RSSeatMapPriceConnectionID	Identity	N	Primary key, always unique
RSSeatMapID	Integer	N	FK link to RSSeatMaps.RSSeatMapID. This is the ID of the reserved seating map.
RSSeatMapPriceID	Integer	N	FK link to RSSeatMapPrices.RSSeatMapPriceID. This is the ID of the price configuration plan.

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatMapPriceConnectionID	P	RSSeatMapPriceConnectionID	Primary Key.

### 18.43 RSSeatMaps

The RSSeatMaps table contains a header record for reserved seating maps.

#### Columns

Column	Type	Allow Nulls	Description
RSSeatMapID	Identity	N	Primary key, always unique.
Name	Char(40)	N	Name to identify the seat map
Description	Text	Y	Can be used to store a description of seat map
SeatCount	Integer	Y	The total number of seats defined in the seat map; maintained by system and stored here for ease of reference; calculated as: select sum(seatCount) from RSSections where RSSeatMapID = n

#### Indexes

Name	Kind	Columns	Purpose
PKRSSeatMapsRSSeatMapID	P	RSSeatMapID	Primary Key.

## 18.44 RSSeatPrices

The RSSeatPrices table can be used to relate a price plan with a specific seat of a reserved seating map. If a record exists in this table, it is the final price override for a seat. The parent tables to this are RSRowPrices, RSSectionPrices, and RSSeatMapPrices.

### Columns

Column	Type	Allow Nulls	Description
RSSeatPriceID	Identity	N	Primary key, always unique
RSSeatMapPriceID	Integer	N	FK link to RSSeatMapPrices.RSSeatMapPriceID. This is the ID of the reserved seating map price record, the parent to this record.
SectionName	Char(10)	N	The name character value that is set by the user during the configuration process.
RowName	Char(10)	N	The name character value that is set by the user during the configuration process.
SeatName	Char(10)	N	The name character value that is set by the user during the configuration process.
RSPricelD	Integer	Y	FK link to RSPrices.RSPricelD. This is the ID of the Price Plan that will be used for this reserved seating seat.
SectionID	Integer	N	FK to the RSSections table
RowID	Integer	N	FK to RSRows table
SeatID	Integer	N	FK to RSSeats table

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatPriceID	P	RSSeatPriceID	Primary Key

### 18.45 RSSeatMapPrices

The RSSeatMapPrices table can be used to relate a price plan with an entire reserved seating map. The price plan will be set at this level if it is not overridden by configuration at the section, row, or seat level; the relative tables are RSSectionPrices, RSRowPrices, and RSSeatPrices.

#### Columns

Column	Type	Allow Nulls	Description
RSSeatMapPriceID	Identity	N	Primary key, always unique
Name	varchar(128)	N	A name character value that is set by the user during the configuration process.
RSPricelD	Integer	Y	FK link to RSPrices.RSPricelD. This is the ID of the Price Plan that will be used for this reserved seating map.

#### Indexes

Name	Kind	Columns	Purpose
PKRSSeatMapPriceID	P	RSSeatMapPriceID	Primary Key

## 18.46 RSSeats

The RSSeats table contains a record for each seat configured for a seat map.

### Columns

Column	Type	Allow Nulls	Description
RSSeatID	Identity	N	Primary key, always unique.
RSRowID	Integer	N	Link to RSRows
SeatName	Char(10)	N	Name to distinguish the seat within the row
CoordX	Integer	N	Used to position seat on graphic seat map
CoordY	Integer	N	Used to position seat on graphic seat map
Rank	Integer	N	Used in determining 'best available' seats
Adjacency	Integer	N	Used to determine adjacent seats

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatsRSSeatID	P	RSSeatID	Primary Key.

## 18.47 RSSeatDisplayStatuses

The **RSSeatDisplayStatuses** table defines seat colors and names for Event Seat Selector map editor. The entries in this table are inserted via the upgrade script and cannot be deleted from nor added to. The end-user can only edit the existing entries in the table.

### Columns

Column	Type	Allow Nulls	Description
RSSeatDisplayStatusID	Identity	N	Primary key, always unique
SeatStatus	Int	N	Value corresponding to the TEventSeatGraphicStatus set <sup>1</sup>
DisplayName	VarChar(20)	N	User defined name for the status above
Description	VarChar(128)	N	Description of the Graphic Status
Color	Int	N	User selectable seat color (used for display in the editor)

### Indexes

Name	Kind	Columns	Purpose
PKRSSeatDisplayStatusID	P	RSSeatDisplayStatusID	Primary Key

<sup>1</sup> **TEventSeatGraphicStatus** type descriptive Values

GraphicStatus	Gateway Constant Name	Description
Available	gsAvailable	Currently available for attempted selection
Session Reserved	gsSessionReserved	Locally reserved by the user DURING THE CURRENT SESSION
Locally Reserved	gsLocallyReserved	Locally reserved by the user
Locally Exchanging	gsLocallyExchanging	Locally reserved, but will be exchanged for new seat
Remotely Reserved	gsRemotelyReserved	Remotely reserved
Purchased	gsPurchased	Purchased by someone
Held	gsHeld	On hold, current agent can release
Unavailable	gsUnavailable	Unavailable for some reason (unknown SeatStatus)
Filtered	gsFiltered	Filtered from view

## 18.48 RSSectionMaps

The RSSectionMaps table links child resources to sections with the specified seat map.

### Columns

Column	Type	Allow Nulls	Description
RSSectionMapID	Identity	N	Primary key, always unique
ResourceID	Integer	N	FK to RMResources table, specifies the child resource that this entry points to
RSResourceMapID	Integer	N	FK to RSResourceMaps table, specifies the entry in the RSResourceMaps table that defines the association between the parent resource and the seat map that contains the section referenced in this entry
RSSeatMapID	Integer	N	FK to the RSSeatMaps table, points to the specific seat map that contains the section referenced in this entry
RSSectionID	Integer	N	FK to the RSSections table, points to the specific section to link this child resource to

### Indexes

Name	Kind	Columns	Purpose
PKRSSectionMapsRSSectionMapID	P	RSSectionMapID	Primary Key

## 18.49 RSSectionPrices

The RSSectionPrices table can be used to relate a price plan with a section of a reserved seating map. The price plan will be set at this level if it is not overridden by configuration at the row or seat level; the relative tables are RSRowPrices and RSSeatPrices. The parent table to this is RSSeatMapPrices.

### Columns

Column	Type	Allow Nulls	Description
RSSectionPriceID	Identity	N	Primary key, always unique
RSSeatMapPriceID	Integer	N	FK link to RSSeatMapPrices.RSSeatMapPriceID. This is the ID of the reserved seating map price record, the parent to this record.
SectionName	Char(10)	N	The name character value that is set by the user during the configuration process.
RSPriceID	Integer	Y	FK link to RSPrices.RSPriceID. This is the ID of the Price Plan that will be used for this reserved seating section.
SectionID	Integer	N	FK to the RSSections table

### Indexes

Name	Kind	Columns	Purpose
PKRSSectionPriceID	P	RSSectionPriceID	Primary Key

## 18.50 RSSections

The **RSSections** is a child table to **RSSeatMaps**, and contains a record for each section configured for a seat map.

### Columns

Column	Type	Allow Nulls	Description
RSSectionID	Identity	N	Primary key, always unique.
RSSeatMapID	Integer	N	FK link to <b>RSSeatMap</b>
SectionName	Char(10)	N	Name to distinguish the Section within the map
SeatOrientation	Integer	N	Indicates the numbering scheme for rows and seats: LF, RF, etc.
Rank	Integer	Y	Used to determine the next section when looking at the best available seat options.
PictureID	Integer	Y	FK to a record the <b>Pictures</b> table that holds an image displaying a view of the stage from the section.

### Indexes

Name	Kind	Columns	Purpose
PKRSSectionsRSSectionID	P	RSSectionID	Primary Key.

## 18.51 RSTests

The RSTests table contains test definitions for the load test application to run.

### Columns

Column	Type	Allow Nulls	Description
RSTestID	Int	N	Primary key, always unique
Name	Varchar(100)	N	Name of this test
EventID	Int	N	FK to RMEvents, EventID used for the test
MaxTransactionSeats	Int	N	Maximum # of seats per transaction
TransactionInterval	Int	N	Interval (in milliseconds) between each transaction
MaxTransactionCount	Int	N	Maximum number of transaction per test
AutoStart	Bit	N	Controls whether the test starts automatically when the event goes on sale
PerformHoldRelease	Bit	N	Perform hold/release combinations during the test
PerformHoldPurchase	Bit	N	Perform hold/purchase combinations during the test

### Indexes

Name	Kind	Columns	Purpose
PKRSTestID	P	RSTestID	Primary Key

## 18.52 RSTestGroups

The RSTestGroups table contains the header records for a set of one to many rows in the RSTestGroupDetails table. This table allows you to define groups of tests to be run in the RS Load Test application.

### Columns

Column	Type	Allow Nulls	Description
RSTestGroupId	Int	N	Primary key, always unique
Name	Varchar(100)	N	Name of this test

### Indexes

Name	Kind	Columns	Purpose
PKRSTestGroupId	P	RSTestGroupId	Primary Key

### 18.53 RSTestGroupDetails

The RSTestGroupDetails table contains one to many rows associated by the RSTestGroupID column value.

#### Columns

Column	Type	Allow Nulls	Description
RSTestGroupDetailID	Int	N	Primary key, always unique
RSTestGroupID	Int	N	FK to RSTestGroups table
RSTestID	Int	N	FK to RSTests table

#### Indexes

Name	Kind	Columns	Purpose
PKRSTestGroupDetailID	P	RSTestGroupDetailID	Primary Key

## 18.54 RSTestTransactions

The RSTestTransactions table contains all transactions for a given load test. These entries are linked to the RSTests table by the RSTestID column.

### Columns

Column	Type	Allow Nulls	Description
RSTestTransactionID	Int	N	Primary key, always unique
RSTestID	Int	N	FK to RSTests table
TestIndex	Int	N	Index of the load test within a load test list
TestAction	Int	N	The action to apply to the seats, hold, release, purchase, etc... <sup>1</sup>
SeatCount	Int	N	The number of seats to reserve
RSSeatID	Int	N	FK to RSSeats; the seat to reserve

### Indexes

Name	Kind	Columns	Purpose
PKRSTestTransactionID	P	RSTestTransactionID	Primary Key

<sup>1</sup> TestAction Values

Value	Gateway Constant Name	Description
0	aaReserveBestAvailable	Place seats into "transaction" state using Get Best Available algorithm
1	aaReserveSeat	Reserve a specific seat
2	aaGetSpecifiedSeats	Get a specific set of seats (not yet implemented)
3	aaPurchaseSeats	Purchase transaction seats
4	aaReleaseSeats	Release transaction seats

## 18.55 SeatAssignmentImports

The SeatAssignmentImports table contains the seat assignment information of an event ticket. The Section, Row and Seat assignment information is assigned by the third party system. The seat assignment data is imported from the third party system by eGalaxy server.

### Columns

Column	Type	Allow Nulls	Description
SeatAssignmentImportID	Int	N	Primary key, always unique
OrderLineID	Int	Y	OrderLineID of the event ticket that is part of an order and was imported from third party system. FK reference to OrderLines.OrderLineID column
VisualID	Char(40)	Y	VisualID of the event ticket imported from third party system. FK reference to Tickets.VisualID column
Section	Varchar(40)	Y	Section of seat assignment
Row	Varchar(40)	Y	Row of seat assignment
Seat	Varchar(40)	Y	Seat of seat assignment

### Indexes

Name	Kind	Columns	Purpose
PKSeatAssignmentImportID	P	SeatAssignmentImportID	Primary Key.
IXSeatAssignmentImpOrderLineID		OrderLineID	Used by query to get data by OrderLineID
IXSeatAssignmentImpVisualID		VisualID	Used by query to get data by VisualID
IXSeatAssignmentImpSecRowSeat		Section, Row, Seat	Used by query to get data by Section, Row and Seat

## 19 SIAE

**19.1 JnlSIAE**

The JnlSIAE table contains information from SIAE records in the journal. The JnlCodeID values for the SIAE entries are: 1010, 1011, 1012, 1013, 1020, 1021, and 1022.

Column	Type	Allow Nulls	Description
JnlSIAEID	Integer	No	Primary key, always unique.
JNLTranID	Integer	Yes	Foreign key to JnlHeaders.JnlTranID
JNLTicketID	Integer	Yes	Foreign key to JnlTickets.JnlDetailID
FiscalSeal	VarChar(23)	No	The fiscal seal generated by the SIAE smart card device for the sale.
ProgressiveNo	Integer	No	The progressive number generated with the fiscal seal.
SmartCardDeviceID	VarChar(9)	No	The device ID (serial number) from the SIAE smart card device that generated the fiscal seal for the sale of the ticket.
FiscalSealDatetime	DateTime	No	The date and time the fiscal seal was generated for the sale.
SystemOwnerTaxID	VarChar(16)	No	Tax ID for the system owner. From SIAEOptions.SystemOwnerTaxID.
EventName	VarChar(30)	No	Name of the SIAEEvent that the ticket is for. From SIAEEvents.Name.
EventType <sup>1</sup>	VarChar(10)	No	Number of EventType. From SIAEEvents.EventType.
EventStartTime	DateTime	No	Start date and time from the ticket's event. From SIAEEvents.StartDateTime.
SIAEOrganizerID	Integer	No	The ID of the SIAE organizer. Foreign key to SIAEEventOrganizers.SIAEEventOrganizerID
VenueTaxID	VarChar(20)	No	The tax ID of the SIAE venue for the ticket. Foreign key to SIAEVenues.TaxID.
VenueType	VarChar(5)	No	The venue type configured on the item. From SIAEItems.VenueType.
EventID	Integer	No	The event ID of the SIAE event configured on the item. Foreign key to SIAEEvents.SIAEEventID.
ReductionCode	VarChar(5)	No	Reduction code configured on the item. Foreign key to SIAEReductionCodes.Code.
ReductionName	VarChar(30)	No	Name of the ticket from the item record. From Items.Name.
TaxType <sup>2</sup>	VarChar(5)	No	SIAE tax type for the ticket.
BaseTaxType <sup>3</sup>	VarChar(5)	No	SIAE base tax type for the ticket.
TurnType <sup>4</sup>	VarChar(5)	No	SIAE turn type for the ticket.
PLU	Char(20)	No	PLU of the ticket. Foreign key to Items.PLU.
AccessCode	Integer	No	Access code for the ticket. Foreign key to AccessCodes.AccessCode.
SIAEVenuelD	Integer	No	Foreign key to SIAEVenues.SIAEVenuelD. The venue ID of the SIAE venue configured on the item.
AbsoluteItemCounter	Integer	Yes	A unique identifier within the Galaxy system for the sale of tickets that do not have a fixed date. Used by reports. This identifier is obtained from the SIAEAbsoluteItemCounters table.
RMEventStartDate	DateTime	Yes	The start date of the resource managed event. This is the date portion from RMEvents.StartDateTime.
VoidReasonCode	VarChar(10)	Yes	The SIAE reason code for the void or return. Foreign key to SIAEVoidReasons.Code.
OriginalDateSold	DateTime	Yes	The DateSold value from the JnlTicket record for original sale transaction when this is a void or a return.
OriginalFiscalSeal	VarChar(23)	Yes	The fiscal seal from the original sale of a ticket when this is a void or return.
OriginalProgressiveNo	Integer	Yes	The progressive number for the fiscal seal obtained from the original sale transaction when this is a void or return.
OriginalSmartCardDeviceID	VarChar(9)	Yes	The ID of the smart card device used to generate the fiscal seal for the original sale transaction when this is a void or return.
EventOrganizerTaxID	VarChar(50)	Yes	The tax ID for the event organizer that sold the ticket. From SIAEEventOrganizers.TaxID.
EventEndDateTime	DateTime	Yes	The end date and time of the SIAE event. From SIAEEvents.EndDateTime.
ReductionCodeDescription	VarChar(100)	Yes	The description of the SIAE reduction code. From SIAEReductionCodes.Description.
ValidVoidEndDate	DateTime	Yes	The last date the SIAE ticket can be voided or returned. This value is calculated at the time of sale.
SoftwareVendorCertificationID	Char(8)	Yes	Unique identifier of the system in SIAE. From SIAEOptions.SoftwareVendorCertificationID.

**Indexes**

Name	Kind	Columns	Purpose
PKJnlSIAEJnlSIAEID	P	JnlSIAEID	Primary key - Unique ID in the table.
IXJnlSIAEJnlTicketID	A	JnlTicketID	Index to aid lookups of SIAE information for a specific ticket. A visual ID lookup is performed on the JnlTickets table, and from that record, the SIAE information is obtained.
IXJnlSIAEPLU	A	PLU	Index to aid lookups of SIAE information by PLU. Used for reports.

**1 EventType Values**

Value	Description
71	Show (Mostre)
77	Amusement or water park (Parchi divertimento e acquatici)

**2 TaxType Values**

Value	Description
'A'	Abbonamento. This is for tickets that are valid for multiple dates.
'O'	Open. This is for tickets that have one use, but are valid on any date.
'T'	Titolo. This is for tickets that are for a single fixed-date.

**3 BaseTaxType Values**

Value	Description
'F'	In Fattura. This is for invoiced tickets.
'N'	Normale. This is for tickets are paid for at the time of sale.

**4 TurnType Values**

--	--

Value	Description
'F'	Fisso. This is for tickets with fixed dates.
'L'	Libero. This is for tickets without a fixed date.

## 19.2 SIAEAbsoluteItemCounters

The SIAEAbsoluteItemCounters table contains counter values that can be associated with a PLU string.

Column	Type	Allow Nulls	Description
PLU	VarChar(20)	No	PLU string value to use for a counter. This is also used with non-PLU strings to generate a counter for things such as SIAE voids. For a void, PLU is "VOID".
CountValue	Integer	No	A counter value that is associated with a PLU. Every time a counter is needed for the PLU string, the value in this field is incremented by 1.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEAbsoluteItemCountersPLU	P	PLU	Primary key - Unique ID in the table. Counter is stored per PLU, so PLU is unique.

### 19.3 SIAECalendarDetails

The SIAECalendarDetails table contains a dates for the SIAECalendarHeaders table.

Column	Type	Allow Nulls	Description
SIAECalendarDetailID	Integer	No	Primary key, always unique.
SIAECalendarID	Integer	No	Foreign key to SIAECalendarHeaders.SIAECalendarID.
Date	DateTime	No	Date for the calendar

#### Indexes

Name	Kind	Columns	Purpose
PKSIAECalendarDetailID	P	SIAECalendarDetailID	Primary key - Unique ID in the table.
IXSIAECalDtlsSIAECalendarID	A	SIAECalendarID	Used to speed queries loading the calendar details by the calendar ID (SIAECalendarHeaders.SIAECalendarID)

## 19.4 SIAECalendarHeaders

The SIAECalendarHeaders table contains a group of dates defined for SIAE.

Column	Type	Allow Nulls	Description
SIAECalendarUniqueID	Integer	No	Primary key, always unique.
SIAECalendarID	Integer	No	Unique identifier for a SIAE calendar. Used as a reference locally and in SQL.
Name	VarChar(50)	No	Name of the SIAE calendar.
CalendarType <sup>1</sup>	Integer	Yes	Type of calendar for SIAE.
Active	Bit	Yes	Indicates if the calendar is currently valid for use in the system.

### Indexes

Name	Kind	Columns	Purpose
PKSIAECalendarUniqueID	P	SIAECalendarUniqueID	Primary key - Unique ID in the table.
IXSIAECalHdrsSIAECalendarID	A	SIAECalendarID	Used to speed queries loading the calendar header by SIAECalendarID, which is the most common method for referencing a calendar.

### <sup>1</sup> CalendarType Values

Value	Gateway Constant Name	Description
0	SIAE_CALENDARTYPE_VOIDBUSINESSDAYS	Void business days calendar type.

## 19.5 SIAEEventOrganizers

The SIAEEventOrganizers table contains information about a SIAE event organizer.

Column	Type	Allow Nulls	Description
SIAEEventOrganizerUniqueID	Integer	No	Primary key, always unique.
SIAEEventOrganizerID	Integer	No	Unique identifier for a SIAE Event Organizer. Used as a reference locally and in SQL.
Name	VarChar(100)	No	Name of the SIAE event organizer.
TaxID	VarChar(50)	No	Tax ID for the SIAE event organizer.
OrganizerType <sup>1</sup>	Integer	Y	Type of event organizer.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEEventOrganizerUniqueID	P	SIAEEventOrganizerUniqueID	Primary key.

### <sup>1</sup> OrganizerType Values

Value	Gateway Constant Name	Description
0	ORGANIZER_TYPE_GENERIC	Generic (Generico)
1	ORGANIZER_TYPE_ESSAY	Essay
2	ORGANIZER_TYPE_PARISH	Parish (Parrocchiale)

## 19.6 SIAEEvents

The SIAEEvents table contains information about SIAE events.

Column	Type	Allow Nulls	Description
SIAEEventUniqueID	Integer	No	Primary key, always unique.
SIAEEventID	Integer	No	Unique identifier for a SIAE event. Used as a reference locally and in SQL.
Name	VarChar(100)	No	Name of the SIAE event.
EventType <sup>1</sup>	Char(10)	No	Type of SIAE event.
SystemOwnerName	VarChar(100)	No	Name of the system owner for this event.
SystemOwnerTaxID	VarChar(50)	No	Tax ID of the system owner for the event.
SIAEOrganizer	Integer	No	Foreign key to SIAEEventOrganizers.SIAEEventOrganizerID.
SIAEVenue	Integer	No	Foreign key to SIAEVenues.SIAEVenueID.
StartTime	DateTime	No	Date and time the event season starts.
EndTime	DateTime	No	Date and time the event season ends.
Active	Bit	No	Indicates if the event is currently active and events can be sold for it.
EventClass <sup>2</sup>	Char(5)	No	Class of event (currently always Spettacolo).
VoidBusinessDaysCalendarID	Integer	Yes	Calendar defining all valid days for the event. This is used to calculate the last date that a ticket can be voided or returned. Foreign key to SIAECalendarHeaders.SIAECalendarID.
Author	NVarChar(100)	Yes	Author of the event (applicable to certain event types)
FilmNationality	NVarChar(2)	Yes	Nationality of the film. Applies only to cinema events.
Company	NVarChar(100)	Yes	Company/Performer for the event (applicable to certain event types)

### Indexes

Name	Kind	Columns	Purpose
PKSIAEEventsSIAEEventUniqueID	P	SIAEEventUniqueID	Primary key.

### <sup>1</sup> EventType Values

Value	Description
1	Cinema
45	Teatro prosa
46	Teatro prosa dialettale
47	Teatro repertorio napoletano
48	Teatro lirico
49	Balletto classico e moderno
50	Operetta
51	Riviste - Commedie musicali
52	Concerti classici
53	Concerti musica leggera
54	Varieta
55	Burattini - Marionette
56	Recitals letterari
57	Concerti bandistici - corali
58	Concerti jazz
59	Concerti di danza
60	Ballo con musica dal vivo
65	Concertini con musica dal vivo
70	Fiere
71	Show (Mostre)
77	Amusement or water park (Parchi divertimento e acquatici)

### <sup>2</sup> EventClass Values

Value	Description
'S'	Spectacle (Spettacolo)
'I'	Entertainment (Intrattenimenti)

## 19.7 SIAEItems

The SIAEItems table contains SIAE information for an item record.

Column	Type	Allow Nulls	Description
SIAEItemID	Integer	N	Primary key, always unique.
PLU	Char(20)	N	Foreign key to Items.PLU.
SIAEEventID	Integer	N	Foreign key to SIAEEvents.SIAEEventID. The event that the item is associated to.
VenueType	Char(5)	N	The SIAE venue type of the item.
ReductionCode	Char(5)	N	The SIAE reduction code for the item. Foreign key to SIAEReductionCodes.Code.
TaxType <sup>1</sup>	Char(5)	N	SIAE tax type for the item.
BaseTaxType <sup>2</sup>	Char(5)	N	SIAE base tax type for the item.
TurnType <sup>3</sup>	Char(5)	N	SIAE turn type for the item.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEItemsSIAEItemID	P	SIAEItemID	Primary key.

#### <sup>1</sup> TaxType Values

Value	Description
'A'	Abbonamento. This is for tickets that are valid for multiple dates.
'O'	Open. This is for tickets that have one use, but are valid on any date.
'T'	Titolo. This is for tickets that are for a single fixed-date.

#### <sup>2</sup> BaseTaxType Values

Value	Description
'F'	In Fattura. This is for invoiced tickets.
'N'	Normale. This is for tickets are paid for at the time of sale.

#### <sup>3</sup> TurnType Values

Value	Description
'F'	Fisso. This is for tickets with fixed dates.
'L'	Libero. This is for tickets without a fixed date.

## 19.8 SIAEOptions

The SIAEOptions table contains general SIAE system configuration settings. There should only be one record in this table.

Column	Type	Allow Nulls	Description
SIAEOptionID	Integer	No	Primary key, always unique.
SystemOwnerName	VarChar(255)	No	Name of the system owner.
SystemOwnerTaxID	Char(16)	No	Tax ID for the system owner.
SoftwareVendorCertificationID	Char(8)	No	Unique identifier of the system in SIAE. This value is populated automatically by the smart card, and cannot be changed once it is set. (Codice Sistema)
SenderEmail	VarChar(255)	No	Email address of the sender for XML report emails.
RecipientEmail	VarChar(255)	No	Email address of the recipient for XML report emails.
CertificationNumber	VarChar(255)	No	SIAE system certification number.
CertificationDate	DateTime	No	Date of SIAE system certification.
LogDirectory	VarChar(200)	No	Directory where XML logs and reports are stored when generated.
EventCancelFOP	Integer	Yes	FOP to use when journalizing the SIAE Event Cancellation transaction
EventCancelNode	Integer	Yes	Node number to use when journalizing the SIAE Event Cancellation transaction

### Indexes

Name	Kind	Columns	Purpose
PKSIAEOptionsSIAEOptionID	P	SIAEOptionID	Primary key.

## 19.9 SIAEReductionCodes

The SIAEReductionCodes table contains the SIAE reduction codes that give tax identifiers for items in the system.

Column	Type	Allow Nulls	Description
SIAEReductionCodeID	Integer	No	Primary key, always unique.
Code	Char(5)	No	Code value for the reduction code.
Description	VarChar(100)	Yes	Description of the reduction code.
IsStatic	Bit	No	Indicates if the reduction code is a standard one that was added automatically, or if it was added by the user. If this value is 1, the reduction code was added automatically.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEReductionCodeID	P	SIAEReductionCodeID	Primary key.

## 19.10 SIAEVenueDetails

The SIAEVenueDetails table contains settings that can be configured on a venue that can be different per venue type. This includes capacity values and amount of tax on regular tickets.

Column	Type	Allow Nulls	Description
SIAEVenueDetailID	Integer	No	Primary key, always unique.
SIAEVenueID	Integer	No	The venue the detail is for. Foreign key to SIAEVenues.SIAEVenueID.
VenueType	Char(5)	No	The venue type (or section) being defined. There will be a different record for each venue type that has data.
Capacity	Integer	No	The capacity for the given venue and venue type.
ExcessComplimentaryTax	Money	Yes	The excess complimentary tax on a regular ticket for the given venue and venue type.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEVenueDetailID	P	SIAEVenueDetailID	Primary key - Unique ID in the table.
IXSIAEVenueIDVenueType	A	SIAEVenueID, VenueType	Index to aid loading of venue details by venue ID and venue type. It is common to try to load venue details for a specific venue and venue type.

## 19.11 SIAEVenues

The SIAEVenues table contains information about SIAE venues.

Column	Type	Allow Nulls	Description
SIAEVenueUniqueID	Integer	No	Primary key, always unique.
SIAEVenueID	Integer	No	Unique identifier for a SIAE venue. Used as a reference locally and in SQL.
Name	VarChar(100)	No	Name of the SIAE venue.
TaxID	VarChar(50)	No	Tax ID for the SIAE venue.
AddressID	Integer	No	Address of the venue. Foreign key to Addresses.AddressID

### Indexes

Name	Kind	Columns	Purpose
PKSIAEVenuesSIAEVenueUniqueID	P	SIAEVenueUniqueID	Primary key.

## 19.12 SIAEVenueUsage

The SIAEVenueUsage table contains a mapping between a usage record and a SIAE venue name. When a usage is imported from Handshake with the Usage Import service, the message from Handshake contains a venue name that is mapped to a SIAEVenue record in Galaxy. This mapping is used to get a SIAEVenueID, which is stored with the usage record.

Column	Type	Allow Nulls	Description
SIAEVenueUsageId	Integer	No	Primary key, always unique.
UsageId	Integer	No	Foreign key to Usage.UsageId.
VenueName	VarChar(256)	Yes	The name of the SIAE venue. From SIAEVenues.Name.

### Indexes

Name	Kind	Columns	Purpose
PKSIAEVenueUsageId	P	SIAEVenueUsageId	Primary key - Unique ID in the table.
IXSIAEVenueUsageIndex	A	UsageId, VenueName	Index for loading record by usage ID and venue name.

### 19.13 SIAEVirtualTickets

The SIAEVirtualTickets table contains SIAE information from ticket scans in the Skidata Handshake application for tickets without a fixed date. This data is generated by the Usage Import service.

Column	Type	Allow Nulls	Description
SIAEVirtualTicketID	Integer	No	Primary key, always unique.
VisualID	VarChar(40)	No	Visual ID of the ticket that was scanned in Handshake.
FiscalSeal	VarChar(23)	No	The fiscal seal generated by the SIAE smart card device for the ticket scan.
ProgressiveNo	Integer	No	The progressive number generated with the fiscal seal.
SmartCardDeviceID	VarChar(9)	No	The device ID (serial number) from the SIAE smart card device that generated the fiscal seal for the scan.
FiscalSealDateTime	DateTime	No	The date and time the fiscal seal was generated for the scan.
UseTime	DateTime	No	The date and time of the scan.
VenueID	Integer	No	Foreign key to SIAEVenues.SIAEVenueID. The venue ID of the SIAE venue where the scan took place.
AbsoluteItemCounter	Integer	No	A unique identifier within the Galaxy system for this scan. Used by reports. This identifier is obtained from the SIAEAbsoluteItemCounters table.

#### Indexes

Name	Kind	Columns	Purpose
PKSIAEVirtualTicketID	P	SIAEVirtualTicketID	Primary key.

## 19.14 SIAEVoidReasons

The SIAEVoidReasons table contains information about SIAE void reasons, which are required for all voids and returns in any SIAE system.

Column	Type	Allow Nulls	Description
SIAEVoidReasonID	Integer	No	Primary key, always unique.
Code	VarChar(10)	No	The code (supplied by SIAE) for the void reason
Reason	VarChar(100)	No	Text description of the void reason (also supplied by SIAE).

### Indexes

Name	Kind	Columns	Purpose
PKSIAEVoidReasonID	P	SIAEVoidReasonID	Primary key - Unique ID in the table.

## 19.15 SIAEXMLReportTrack

The SIAEXMLReportTrack table contains information about how many times an XML report was generated for a given date. It is used for file name counters, and for indicating if the report is the original, or is a replacement for the original.

Column	Type	Allow Nulls	Description
SIAEXMLReportTrackID	Integer	No	Primary key, always unique.
ReportPeriod <sup>1</sup>	Integer	No	Time period over which the report was run.
ReportDate	VarChar(10)	No	Date or month the report was run for.
ReportType <sup>2</sup>	VarChar(3)	No	Identifies the report type (used in the naming of the XML report files)
GeneratedCount	Integer	No	Number of times the report has been run for the given date

### Indexes

Name	Kind	Columns	Purpose
PKSIAEXMLReportTrackID	P	SIAEXMLReportTrackID	Primary key - Unique ID in the table.
IXSIAEXRTPeriodDateType	A	ReportPeriod, ReportDate, ReportType	Index to aid loading and updating the counter by report period, date, and type.

### <sup>1</sup> ReportPeriod Values

Value	Gateway Constant Name	Description
0	SIAE_DAILY_REPORT	Daily report
1	SIAE_MONTHLY_ALL_REPORT	Monthly report (for the entire month)

### <sup>2</sup> ReportType Values

Value	Description
'LOG'	Log report type
'RPG'	Daily report type
'RPM'	Monthly report type

## 20 Smart Upsell

Smart Upsell allows changing upsell and upgrade options dynamically based on external conditions.

## 20.1 SmartUpsellSalesGoals

This table stores Smart Upsell sales goals. The details of the goals are stored in SmartUpsellSalesGoalDetails.

### Columns

Column	Type	Allow Nulls	Description
SmartUpsellSalesGoalID	Int	N	Primary key, always unique. Identity column.
Description	Nvarchar(100)	N	The description of the sales goal.
ItemGroupID	Int	N	The ID of the GxItemGroup that will be used for calculations.
GoalSalesCountVariableName	Nvarchar(100)	N	The name of the external variable that will be created or updated to store the sales goal count.
GoalsMetVariableName	Nvarchar(100)	N	The name of the external variable that will be created or updated to store if the sales goal has been met.
SalesCountVariableName	Nvarchar(100)	N	The name of the external variable that will be created or updated to store the actual sales count.

### Indexes

Name	Kind	Columns	Purpose
PKSmartUpsellSalesGoalID	P	SmartUpsellSalesGoalID	Primary Key.

## 20.2 SmartUpsellSalesGoalDetails

This table stores Smart Upsell sales goal details. The header information for the goals is stored in SmartUpsellSalesGoals.

### Columns

Column	Type	Allow Nulls	Description
SmartUpsellSalesGoalDetailID	Int	N	Primary key, always unique. Identity column.
SmartUpsellSalesGoalID	Int	N	Foreign key to SmartUpsellSalesGoals.SmartUpsellSalesGoalID, specifying the sales goal that this detail applies to.
GoalSalesCount	Int	N	The sales count goal for the specified date range.
GoalStartDate	DateTime	N	The start of the date range for this goal instance. This represents a fiscal date.
GoalEndDate	DateTime	N	The end of the date range for this goal instance. This represents a fiscal date.

### Indexes

Name	Kind	Columns	Purpose
PKSmartUpsellSalesGoalDetailID	P	SmartUpsellSalesGoalDetailID	Primary Key.

## 20.3 SmartUpsellRules

This table stores Smart Upsell rules.

### Columns

Column	Type	Allow Nulls	Description
SmartUpsellRuleID	Int	N	Primary key, always unique. Identity column.
TargetPLU	Nvarchar(20)	N	The target PLU that will be modified in the ItemUpgrades and UpsellOptions tables.
Sequence	Int	N	A numerical sequence indicating the order rules will be processed for TargetPLU.
Description	Nvarchar(100)	N	The description of the rule.
Inactive	Bit	N	Indicates if the rule will not be processed.
IncludeProducts	Bit	Bit	Y
IncludeUpgrades	Bit	Bit	Y
IncludeUpsellAddOns	Bit	Bit	Y
IncludeUpsellReplacements	Bit	Bit	Y
ConditionScript	Nvarchar(MAX)	N	The condition script that will be evaluated for the rule.
ConditionScriptType	Nvarchar(10)	N	Indicates what scripting language the condition script uses.
ActionEmphasisEnabled	Bit	Bit	Y
ActionEmphasisValue	Int	N	The value that the Emphasis field will be changed to.
ActionHideEnabled	Bit	Bit	Y
ActionHideValue	Bit	Bit	Y
ActionScriptTemplateIDEnabled	Bit	Y	Indicates if the ScriptTemplateID field will be modified when the rule evaluates to true.
ActionScriptTemplateIDValue	Int	N	The value that the ScriptTemplateID field will be changed to.

### Indexes

Name	Kind	Columns	Purpose
PKSmartUpsellRuleID	P	SmartUpsellRuleID	Primary Key.

**21 Voucher**

## 21.1 MarkedVouchers

Column	Type	Allow Nulls	Description
MarkedVoucherID	Int	N	Primary key, Always unique
VoucherID	Int	N	Foreign key to Vouchers.VoucherID
MarkedStatus	Int	N	Indicates state of the voucher marked for pre-print <sup>1</sup>
LockingNodeNo	Int	N	Node number of the POS that locked the voucher for pre-print
PrePrintBatchID	Int	N	

### Indexes

Name	Kind	Columns	Purpose
PKMarkedVouchersMarkdVoucherID	P	MarkedVoucherID	Primary key.

<sup>1</sup> MarkedStatus Values

Value	Description
0	The voucher is locked and waiting to be pre-printed
1	The voucher is currently being pre-printed (still locked)
2	The voucher has been preprinted and unlocked
3	The voucher has been voided during pre-printing

## 21.2 Vouchers

Column	Type	Allow Nulls	Description
VoucherID	Int	N	Primary key, Always unique
BatchID	Int	Y	Foreign key to VoucherBatches.VoucherBatchID
IssuerCode	Char(20)	Y	Link to ExternalAccount field in Customers table
ArrivalDate	DateTime	Y	Scheduled pick-up date for the voucher tickets
ActualArrivalDate	DateTime	Y	Actual pick-up date for the voucher tickets
SerialNumber	Char(30)	N	Voucher serial number
ErrorCode	Int	Y	Error code if error occurred importing this voucher <sup>1</sup>
Status	Int	Y	Status of this voucher <sup>2</sup>
OriginalStatus	Int	Y	Original status of this voucher <sup>2</sup>
IssueDate	DateTime	Y	Date this voucher was issued (purchased)
RecordLock	Int	N	Indicates whether or not this voucher is locked <sup>3</sup>
SalesStatus	Char(1)	Y	Sales status <sup>4</sup>
ReservationNo	Char(16)	Y	Reservation Number
PurposeUse	Char(1)	Y	Purpose of use
TotalOrderedQty	Int	Y	Total quantity of ordered tickets for this voucher
TotalIssuedQty	Int	Y	Total quantity of issued tickets for this voucher
ClassificationCode	Int	Y	Ticket classification (value 1 to 999)

### Indexes

Name	Kind	Columns	Purpose
PKVouchersVoucherID	P	VoucherID	Primary key.
IXVouchersStatus		Status	
IXVouchersArrivalDtIssuedSrlNbr		ArrivalDate, IssuerCode, SerialNumber	

Notes for Vouchers table:

#### <sup>1</sup> ErrorCode Values

Value	Description
0	No error. OK.
1	Duplicate Voucher in a given import batch
2	Dynamic update error. This voucher was already added to the system at a POS. Import would no longer be needed.

#### <sup>2</sup> Status Values

Value	Description
0	Imported (Ok to be redeemed)
1	Issued (Tickets on the voucher have been redeemed)
2	Cancelled (All tickets on this voucher has been cancelled)
3	Dynamically created
4	Pre-Printed
5	Reserved (not currently used)
6	Dynamically created voucher currently being edited

#### <sup>3</sup> RecordLock Values

Value	Description
0	Voucher is not locked
1	Voucher is currently locked
2	Voucher is currently locked for pre-print

#### <sup>4</sup> SalesStatus Values

Value	Description
1	Imported as new voucher
2	Imported as cancelled voucher (not available for redemption).

### 21.3 VoucherDetails

Column	Type	Allow Nulls	Description
VoucherDetailID	Int	N	Primary key, Always unique
VoucherID	Int	N	Foreign key to Vouchers.VoucherID
PLU	Char(20)	N	Foreign key to Items.PLU
OrderedQty	Int	N	Number of tickets with the above PLU this voucher can be used to redeem
IssuedQty	Int	Y	Number of ticket with the above PLU that already have been redeemed
User1	Char(50)	Y	User-specific field 1
User2	Char(50)	Y	User-specific field 2
IssuedTktClass	Char(50)	Y	
IssuedTktCode	Char(50)	Y	

#### Indexes

Name	Kind	Columns	Purpose
PKVoucherDetailVoucherDetailID	P	VoucherDetailID	Primary key.
IXVoucherDetailsVoucherID		VoucherID	

Notes for VoucherDetails table:

1. Current uses for user-specific fields:

1. Travel agent vouchers:
  - User1 - Ticket Number. Value of 1 to 7.
  - User2 - Ticket Classification. Value of 1 to 999.

**21.4 VoucherBatches**

Column	Type	Allow Nulls	Description
VoucherBatchID	Int	N	Primary key, Always unique
ImportDate	DateTime	Y	When this batch of vouchers was imported
BatchName	Char(30)	N	User-definable name of this import batch

**Indexes**

Name	Kind	Columns	Purpose
PKVoucherBatches	VoucherBatchID	P	VoucherBatchID Primary key.

**21.5 VoucherMarketing**

Column	Type	Allow Nulls	Description
VoucherMarketingID	Int	N	
VoucherID	Int	N	
GroupName	Char(30)	Y	
ShopName	Char(20)	Y	
ShopCode	Char(6)	Y	
ShopNameInKanji	Char(30)	Y	
TransferMgmtNo	DateTime	Y	

**Indexes**

Name	Kind	Columns	Purpose
PKVoucherMrktngVoucherMrktngID	P	VoucherMarketingID	Primary key.

## 22 Web Store

These tables are used by the eGalaxy Web Store exclusively.

## 22.1 CustContactsLogin

### Columns

Column	Type	Allow Nulls	Description
CustContactLoginID	int	N	Primary key, always unique (MS SQL Identity column)
CustContactID	int	Y	Foreign key to CustContacts.CustContactID
Email	NVarChar(150)	Y	
UserPassword	Char(20)	Y	

### Indexes

Name	Kind	Columns	Purpose
PKCustContactLoginID	P	CustConctactLoginID	Primary Key.

## 22.2 EditedBillingInfo

### Columns

Column	Type	Allow Nulls	Description
ExternalID	Char(20)	N	
OriginalName	Varchar(250)	Y	
NewName	Varchar(250)	Y	
LastUpdate	DateTime	Y	

### Indexes

Name	Kind	Columns	Purpose

### 22.3 OrderCreditCardLog

This table keeps a log of each attempt to authorize a credit card when checking out on the web store.

#### Columns

Column	Type	Allow Nulls	Description
ID	Int	N	Primary key, always unique (MS SQL Identity column)
ExternalOrderID	VarChar(30)	Y	Web Store Order ID
RequestXML	Text	Y	XML that is sent to credit card processor
ResponseXML	Text	Y	XML received from credit card processor
AuthCode	VarChar(128)	Y	Authorization code received
CCErrCode	Int	Y	Credit card error code received
DateSubmitted	DateTime	Y	Date of transaction

#### Indexes

Name	Kind	Columns	Purpose
PKOrderCreditCardLogID	P	ID	Primary Key.

## 22.4 RejectedOrderErrors

### Columns

Column	Type	Allow Nulls	Description
ID	Integer	N	Primary key, SQL Identity column
OrderID	Varchar(25)	Y	
ErrorCode	Char(5)	Y	
ErrorText	Text	Y	Error text

### Indexes

Name	Kind	Columns	Purpose
PKRejectedOrderErrorID	P	ID	Primary Key

## 22.5 RejectedSurveyResultErrors

Table RejectedSurveyResultErrors contains the errors for the surveys rejected by eGalaxy. The table has ErrorCode and ErrorText columns, which contains the details of the error from eGalaxy. This table is used in the web store's database.

### Columns

Column	Type	Allow Nulls	Description
RejectedSurveyResultErrorID	Integer	N	Primary key, gateway counter
SurveyResultID	Integer	N	FK reference to SurveyResults table
ErrorCode	Integer	N	Error code
ErrorText	Text	Y	Error text

### Indexes

Name	Kind	Columns	Purpose
PKRejectedSurveyResultErrorID	P	RejectedSurveyResultErrorID	Primary Key, gateway counter
IXRejSurResErrorsSurvResultID		SurveyResultID	Used by the query to get errors for a given SurveyResultID

## 22.6 RetrievePrices

This table stores prices returned from CartTrans until they are displayed on the web store

### Columns

Column	Type	Allow Nulls	Description
ID	Int	N	Primary key, always unique (MS SQL Identity Column)
ExternalOrderID	Varchar(30)	Y	Web Store Order ID
PLU	Varchar(20)	N	PLU of item
Price	Money	N	Price of item received from CartTrans
Discount	Money	N	Discount off item price received from CartTrans
Tax	Money	N	Tax on item received from CartTrans
LastUpdate	DateTime	N	Date of transaction

### Indexes

Name	Kind	Columns	Purpose
PKRetrievePriceID	P	ID	Primary Key.

**22.7 Sched1**

Sched 1 is a web store, transportation table

**Columns**

Column	Type	Allow Nulls	Description
Sched1ID	Int	No	Primary Key
EffDate	DateTime	No	Date this schedule segment becomes effective
ExpDate	DateTime	Yes	Expiration date; null or zero means the schedule has no expiration date
TableNo	Char(4)	Yes	Table number in the Russell's Guide
Carrier	Char(4)	No	Refers to Carriers.Carrier
Sched	Char(4)	No	Schedule number
Seg	SmallInt	No	Segment number
City	Int	Yes	Refers to Cities.Code
Freq	Char(8)	Yes	Service frequency in days YYYYYYYY = MTWTFSSH
Arrive	Datetime	Yes	Time this segment arrives at City
Leave	Datetime	Yes	Time this segment leaves City
RT	SmallInt	Yes	Ride time in minutes
LO	SmallInt	Yes	Layover at this segment in minutes
SegET	SmallInt	Yes	Elapsed time from previous segment in minutes
SegMiles	SmallInt	Yes	Miles from previous segment
MPH	SmallInt	Yes	Miles per hour (used to validate schedule times)
ET	SmallInt	Yes	Cumulative elapsed time from segment 1 in minutes
Miles	SmallInt	Yes	Cumulative miles from segment 1
Marks	Char(8)	Yes	Reference marks; D: Drop Only, F: flag stop, etc.
LastSegment	Bit	Yes	True if this is the last segment for this schedule number
Imported	Bit	Yes	Used when importing TX data to delete missing records

**Indexes**

Name	Kind	Columns	Purpose
PKSched1Sched1ID	PK	Sched1ID	Primary key index
IXUserInterface	IX	Carrier, Sched, EffDate	
IXByCity	IX	City, EffDate, Carrier, Sched, Seg	
IXBySched	IX	EffDate, Carrier, Sched, Seg	
IXBySeg	IX	Carrier, Sched, Seg, EffDate	

## 22.8 SessionAssign

### Columns

Column	Type	Allow Nulls	Description
ID	Int	N	Primary key, always unique, Identity column.
SessionID	NVarChar(125)	N	Session ID
DateTime	SmallDateTime	N	

### Indexes

Name	Kind	Columns	Purpose
PKSessionAssignID	P	ID	Primary Key.

## 22.9 SurveyResults

SurveyResults table contains the header of the survey result. Column Status indicates the status of the survey at any time. This table is used on the web store database to store the survey results.

### Columns

Column	Type	Allow Nulls	Description
SurveyResultID	Int	N	Primary key, always unique
SurveyID	Int	N	FK reference to Surveys table
Status	Int	N	Current status of the survey <sup>1</sup>
PickupDate	DateTime	Y	Date when survey was picked up
OrderID	Char (20)	Y	ID of the order for which the survey was completed

### Indexes

Name	Kind	Columns	Purpose
PKSurveyResultID	P	SurveyResultID	Primary Key.
IXSurveyResultsSurveyID		SurveyID	Used to get results of a given survey
IXSurveyResultsStatus		Status	Used by the query to get surveys by their Status

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	SURVEY_STATUS_NEW	New survey (not picked up yet)
2	SURVEY_STATUS_PICKEDUP	Picked up
3	SURVEY_STATUS_ERROR	Error

## 22.10 SurveyResultDetails

SurveyResultDetails table contains the details of the survey result. Column SurveyResultID is a foreign key to the SurveyResults table, which owns the details in this table. This table is used on the web store database to store the survey results.

### Columns

Column	Type	Allow Nulls	Description
SurveyResultDetailID	Int	N	Primary key, always unique
SurveyResultID	Int	N	FK reference to SurveyResults table
AnswerNumber	Int	N	Answer number
AnswerText	Varchar (40)	Y	Answer text
SurveyFieldID	Int	Y	Foreign key reference to the SurveyFields.SurveyFieldID to help link the Result to its question.

### Indexes

Name	Kind	Columns	Purpose
PKSurveyResultDetailID	P	SurveyResultDetailID	Primary Key.
IXSurveyResDtlsSurveyResultID		SurveyResultID	Used by the query to get survey result details for a given SurveyResultID

## 22.11 WebFares

The WebFares table stores available fares for an origin-destination pair.

### Columns

Column	Type	Allow Nulls	Description
WebFareID	Int	N	Primary key, always unique
FareSet	Int	Y	Group WebFares by departure in WebScheds
FareID	Int	Y	Refers to Fares.FareID
TTTID	Int	Y	Refers to Templates.TemplateID
FullFare	Float	Y	The fare used to calculate ticket types
FareAmout	Float	Y	The (calculated) standard (Adult) fare
ChildFare	Float	Y	Child fare from ticket type fare class CH
StudentFare	Float	Y	Student fare from ticket type fare class ST
SeniorFare	Float	Y	Senior fare from ticket type fare class SC
TripMode	SmallInt	Y	1=OW, 2=RT
Refundable	Bit	Y	0=no refunds, 1=refundable
Freq	Char(7)	Y	1 thru 7 for Mon thru Sun that this fare is valid. For example Where Freq Like '%W%' if traveling on a Wed. Note Freq might be 'MTWR--' or 'M-W-F--'.
FareClass	Char(4)	Y	S=Student, SC=Senior Citizen, M=Military
Restriction	Char(4)	Y	AP=Advance Purchase, EX=Excursion, GO=Depart within {days} etc
Days	SmallInt	Y	Number of days for advance purchase and excursion fares
RestrictionText	VarChar(40)	Y	NO REFUND, AP <days> DAYS, EX <days>, etc
Carrier	Char(4)	Y	Carrier for schedule specific fares
Sched	Char(4)	Y	Schedule for schedule specific fares
Method	Char(1)	Y	V, M, C for Value Mileage or Combined fare.
TariffMiles	Int	Y	The miles used to compute a mileage based fare.
TariffID	Int	Y	Refers to Tariffs.TariffID
TariffName	Varchar(24)	Y	Name of the tariff
TariffEndorsement	Varchar(50)	Y	Tariff description

### Indexes

Name	Kind	Columns	Purpose
PKWebFaresWebFareID	P	WebFareID	Primary Key.
IXWebFaresFareID	I	FareID	Index on FareID
IXWebFaresFareSet	I	FareSet	Index on FareSet

## 22.12 WebScheds

### Columns

Column	Type	Allow Nulls	Description
WebSchedID	Int	N	Primary key, always unique
EffDate	DateTime	Y	First date this departure can be used
ExpDate	DateTime	Y	Last date this departure can be used
TravelDay	SmallInt	Y	1-7 for Mon thru Sun
Origin	Int	Y	Refers to Cities.Code
Destin	Int	Y	Refers to Cities.Code
Carrier	Char(4)	Y	Refers to Cities.Code
Sched	Char(4)	Y	
Leaves	DateTime	Y	Departure time (hh:mm) from the origin
Arrives	DateTime	Y	Arrival time (hh:mm) at the destination
ET	Char(8)	Y	HH:MM or 99h 99m Total trip elapsed time in minutes
Miles	SmallInt	Y	Total trip miles
Breaks	SmallInt	Y	Number of connections
Freq	Char(8)	Y	Days as in YYYYNNYY = M,T,W,T,F,S,S,H(olidays)
FreqText	VarChar(60)	Y	Days as Mon-Fri except Holidays
FareSet	Int	Y	Refers to WebFares.FareSet
Connections	VarChar(128)	Y	City names for transfers

### Indexes

Name	Kind	Columns	Purpose
PKWebSchedsWebSchedID	P	WebSchedID	Primary Key.
IXWebSchedsOriginDestin	I	Origin, Destin	Index on Origin and Destin

## 22.13 WebSchedDetail

### Columns

Column	Type	Allow Nulls	Description
WebSchedDetailID	Int	N	Primary key, always unique
WebSchedID	Int	Y	Detail Key for WebScheds
LegNo	SmallInt	Y	Leg Number
Carrier	Char(4)	Y	Refers to Cities.Code
Sched	Char(4)	Y	
Origin	Int	Y	Refers to Cities.Code
Destin	Int	Y	Refers to Cities.Code
Leaves	DateTime	Y	
Arrives	DateTime	Y	
LayOver	Char(8)	Y	Layover as hh:mm
Stops	SmallInt	Y	Number of stops on this leg
ET	Char(8)	Y	Elapsed time for this leg as hh:mm
Miles	SmallInt	Y	Total miles for this leg
CVMiles	Int	Y	Miles used to calculate the coupon value
Freq	Char(8)	Y	Days the schedule operates
ThruFreq	Char(8)	Y	Days we actually arrive at the next leg

### Indexes

Name	Kind	Columns	Purpose
PKWebSchedDtlWebSchedDetailID	P	WebSchedDetailID	Primary Key.
IXWebSchedDetailWebSchedID	I	WebSchedID	Index on WebSchedID
IXWebSchedDetailOriginDestin	I	Origin, Destin	Index on Origin and Destin

## 22.14 WebStoreErrors

This table stores any errors that may occur on the web store.

### Columns

Column	Type	Allow Nulls	Description
ID	Int	N	Primary key, always unique (MS SQL Identity column)
Page	Varchar(255)	N	Page the error occurred on
Error	Text	N	Error that occurred on the page
Occurred	DateTime	N	Date error occurred

### Indexes

Name	Kind	Columns	Purpose
PKWebStoreErrorID	P	ID	Primary Key.

## 22.15 WSActivityLog

The WSActivityLog table is used to log web application activity such as; admin, workflow, feature or function.

Column	Type	Allow Nulls	Description
WSActivityLogID	Int	N	Primary key, always unique
DateTime	Datetime	N	Date/time stamp for record
Activity	varchar(1024)	N	Free form text field for describing the activity being logged. This could be an administrative function such as "uploaded file header.jpg" or a workflow activity such as "guest names entered" or a general feature/function such as "PLU xyz added to cart".
Application	varchar(200)	Y	Describes the application that the activity record is associated with. Typical values will be: Consumer, Mobile, Reseller
ASPNetSessionID	varchar(200)	Y	The ASP.NET assigned session ID for the current user/session.
RemoteAddress	varchar(50)	Y	The IP of the requesting machine.
HttpUserAgent	varchar(500)	Y	The HttpUserAgent string of the requesting software. i.e. "Mozilla/5.0 (iPad; U; CPU OS 3_2_1 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Mobile/7B405"
WebServerIP	varchar(50)	Y	The IP of the responding web server.

### Indexes

Name	Kind	Columns	Purpose
PKWSActivityLogID	P	WSActivityLogID	Primary Key

## 22.16 WSConfig

Configuration used for Web Store. There can be only one active configuration for a single Web Store. The DefaultConfig column is used for this purpose.

Column	Data Type	Allow Null	Description
WSConfigID	Int	N	Primary Key. Web store configuration ID
Name	VarChar(128)	N	Web store configuration name
DefaultConfig	Bit	N	If true, this is the configuration used in the live Web Store. There can be only one row with this value set to 1.
DefaultLanguage	VarChar(16)	Y	Language to use for the config. If this language is set to Spanish, the base language for the Web Store will be in Spanish.
DefaultCountryCode	char(2)	Y	Country Code for the config. Foreign key to Countries.CountryCode
CartTransWorkingDirectory	VarChar(500)	Y	Cart Tran's SQL Script / working directory
WebPublishingWorkingDirectory	VarChar(500)	Y	WebPublishing SQL Script directory
eGalaxyURL	VarChar(100)	Y	eGalaxy URL, including the port.
eGalaxyTimeout	Int	Y	eGalaxy connection timeout
eGalaxySourceID	VarChar(50)	Y	Source ID to identify Web Store source to eGalaxy. In concept, this is a foreign key to Galaxy database's eGalaxySource.ExternalSourceID. This value must be in eGalaxySources table in order for Web Store request to eGalaxy be processed.
eGalaxyUsername	VarChar(50)	Y	User Name to connect to eGalaxy. This is the name from GxUsers.
eGalaxyPassword	VarChar(50)	Y	Password. Plain text and not encrypted.
StoreOffline	Bit	Y	If true, Store is offline.
AllowAccountUse	Bit	Y	If true, allows the account be created, edited, or allows user to login. This allows customer to create an account to be used in Web Store instead of repeating account information per transaction.
DefaultShippingSameAsBilling	Bit	Y	If true, use the billing address as shipping
ForceShippingSameAsBilling	Bit	Y	If true, always use billing as shipping address
DisplayMiddleNameField	Bit	Y	If true, Web Store displays customer's middle name.
DisplayBillingEmailConfirmationField	Bit	Y	If true, the billing email confirmation is displayed
DisplayShippingEmailField	Bit	Y	If true, the shipping email address is displayed
DisplayOrderNotesField	Bit	Y	If true, Order Notes are displayed
PaymentProvider	VarChar(50)	Y	Payment Provider / Processor for Config
BypassGuestNames	Bit	Y	If true, for Print@Home ticket, do not collect guest name information.
AllowZeroDollarTransactions	Bit	Y	Allow Zero Dollar transactions
AllowAccountPaymentStorage	Bit	Y	If true, allows payment information to be created and stored and edited on an Account.
AllowPassUpdate	Bit	Y	Allow Web Store to update pass info, renew, etc online
EventMonthQueryMax	Int	Y	Months in advance to query for events. Default value is 12.
DisplaySmallCart	Bit	Y	Show small cart. Default value is 1.
TimeZoneID	VarChar(64)	Y	TimeZone for Store used to calculate server DateTime.
PassLookupMask	Int	Y	Store pass lookup fields as bit mask (default=1)
AdminPanelLoginType <sup>1</sup>	Int	Y	Used to determine to use the UserName or UserID from GxUsers when logging into the online admin panel for the web store.
MultipleeGalaxyServers	Bit	Y	Tells Webstore which SQL selects to run for order pickup
DisplayPriceWithTax	Bit	Y	User to enable option to display price with tax - off by default
DisplayLogInInfo	Bit	Y	Show Logged in user name
DisplayLogOffLink	Bit	Y	Show Log Off link
CustomerOrderInfoReadOnly	Bit	Y	Do not allow editing of Show Billing, Shipping and Account information
PaymentPlansRequireLogin	Bit	N	Require customer to login to purchase tickets associated with any Payment Plans. Default is True.
UseUrlPortMapping	Bit	N	Used by web store to determine if the port number should be used in url mapping. Default is True.
RenewalSource	Int	N	1 = PassKind; 2 = SalesChannel; default value is 1
PassPhotoMaxWidth	int	Y	Defines the max width of a photo that can be uploaded
PassPhotoMaxHeight	int	Y	Defines the max height of a photo that can be uploaded
PassPhotoMaxMB	decimal(2,1)	Y	Defines the max size (in megabytes) of a photo that can be uploaded.
PassAllowPhotoUpload	Bit	Y	Used to determine if a photo can be uploaded with a pass.
CutOffTime	Int	Y	CutOffTime in for selecting schedules on the transportation web store
CartItemDescriptionTemplate	Varchar(128)	Y	Holds template for cart item description field
DisplayUserDefinableScheduleLink	Bit	Y	Enables the user definable text option for Schedule link of Schedule column on ETickets page.
DisplayEventEndTime	Bit	Y	Used to determine if the event time should be displayed.
PassDisplayUseAsBilling	Bit	Y	Determine if Use As Billing check box will be shown on Pass Info page, Default is True.
BillingDateOfBirthRequired	Bit	Y	Determine if Date of Birth is shown and required on the Billing page.
BillingDisplayPassContactControl	Bit	Y	Determine if Pass Contact Selection control is shown on Billing page.
HideTaxForTaxIncludedItems	Bit	Y	Determines if the tax on tax-included items is displayed or hidden.
UseEmailForLoginName	Bit	Y	Determines if the emailaddress is used for the login name.
HidePassConfirmationForRenewal	Bit	Y	Determines if the Pass Confirmation page is displayed during pass renewal
ViewItemDescriptionTemplate	Varchar(128)	Y	Override default web store PLU name display

### Indexes

Name	Kind	Columns	Purpose
PKWSConfigID	PK	WSConfigID	Table primary key

<sup>1</sup> AdminPanelLoginType Values

Value	Const Name	Descriptions
0	UserName	Reference to the GxUsers.UserName column

1	UserID	Reference to the GxUsers.UserID column
---	--------	--

## 22.17 WSConfigDetails

The WSConfigDetails table contains web store config settings stored as key/value pairs.

Column	Data Type	Allow Nulls	Description
WSConfigDetailID	Int	N	Primary key, always unique
WSConfigHeaderID	Int	Y	Foreign Key into WSConfigHeaders
Key	Varchar(128)	Y	Key/Property name used to map to web store WebStoreConfig class.
Value	Varchar(128)	Y	Value associated with Key.

### Indexes

Name	Kind	Columns	Purpose
PKWSConfigDetailID	P	WSConfigDetailID	Primary Key.

## 22.18 WSConfigHeaders

The WSConfigHeaders used for Web Store. There can be only one active configuration for a single Web Store. The DefaultConfig column is used for this purpose.

Column	Data Type	Allow Nulls	Description
WSConfigHeaderID	Int	N	Primary key, always unique
Name	Varchar(50)	Y	Name of config set
DefaultConfig	Bit	Y	Makes this config set as the default
Consumer	Bit	Y	Indicates if the configuration is associated with the consumer web store application
PickupAndPublishing	Bit	Y	Indicates if the configuration is associated with the order pickup/publishing web application
Reseller	Bit	Y	Indicates if the configuration is associated with the reseller web application
PassPortal	Bit	Y	Indicates if the configuration is associated with the pass portal web application
GroupSales	Bit	Y	This column is used to determine if the Group Sales module uses this configuration.

### Indexes

Name	Kind	Columns	Purpose
PKWSConfigHeaderID	P	WSConfigHeaderID	Primary Key.

## 22.19 WSKeyValue

This table will allow the web store to store any value that is not part of our current database scheme. For example some payment providers return a session token that needs to be persisted between calls and this gives the web store a place to store and retrieve it.

Column	Data Type	Allows Nulls	Description
WSKeyValueID	Int	N	Primary key, always unique
Type	Int	Y	Type of value stored. 0 = None, 1 = SurePayToken
Key	Varchar(128)	Y	Key used along with Type to retrieve the value
Value	Varchar(1024)	Y	Value associated with Key.

### Indexes

Name	Kind	Columns	Purpose
PKWSKeyValueID	P	WSKeyValueID	Primary Key

## 22.20 WSLocalization

The WSLocalization table contains the text strings for labels, textboxes, etc. that are localizable based on merchant and culture code.

Column	Data Type	Allow Nulls	Description
WSLocalizationID	Int	N	Web store localization ID
Page	Varchar(256)	N	Page link
WSLocalizationGroupID	Int	N	Foreign key to the header table, WSLocalizationGroups.WSLocalizationGroupID
ControlIndex	Int	Y	Web store index
StringValue	nvarchar(max)	N	Web store string value
Module	Int	Y	Indicates which module the localization string value is associated to. <sup>1</sup>
TranslationLanguageID	Int	Y	FK to TranslationLanguages.TranslationLanguageID
StringKey	VarChar(256)	N	Web store key value
WSLocalizationGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKWSLocalizationID	P	WSLocalizationID	Primary Key.
IXWSLocalizationGroupLanguage		WSLocalizationGroupID, TranslationLanguageID	

### <sup>1</sup> Module Values

Value	Gateway Constant Name	Description
0	LOCALIZATION_MODULE_WEBSTORE	Web Store module
1	LOCALIZATION_MODULE_RESELLER	Reseller module
2	LOCALIZATION_MODULE_TRANSPORTATION	Transportation module

## 22.21 WSLocalizationGroups

The purpose of this table is to group a set of WSLocalization entries into one name. This allows multiple Merchants share the same set of WSLocalization.

### Columns

Column	Type	Allow Nulls	Description
WSLocalizationGroupID	Int	N	Primary key, always unique
Name	Varchar(128)	N	A string to identify the Group
StringType <sup>1</sup>	Int	Y	The module the strings are associated with
WSLocalizationGroupGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKWSLocalizationGroupID	P	WSLocalizationGroupID	Primary Key.

### <sup>1</sup> StringType Values

Value	Description
0	Strings are associated with Web Store module
1	Strings are associated with Reseller module
2	Strings are associated with Transportation module
3	Strings are associated with Pass Portal module

## 22.22 WSLocalizationGroupModules

The WSLocalizationGroupModules stores the modules a particular web store localization group contains.

### Columns

Column	Type	Allow Nulls	Description
WSLocalizationGroupModuleID	Int	N	Primary key, always unique
WSLocalizationGroupID	Int	N	Primary key ID for the Localization Group
Module	Int	N	Web store module type <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKWSLocalizationGroupModuleID	P	WSLocalizationGroupModule	Primary Key.
IXWSLocalizationGroupID	I	WSLocalizationGroupID	To facilitate retrieval by WSLocalizationGroupID

<sup>1</sup> Module Values

Value	Gateway Constant Name	Description
0	LOCALIZATION_MODULE_WEBSTORE	Web Store module
1	LOCALIZATION_MODULE_RESELLER	Reseller module
2	LOCALIZATION_MODULE_TRANSPORTATION	Transportation module
3	LOCALIZATION_MODULE_PASS_PORTAL	Pass Portal module

## 22.23 WSLog

A list of log entries from Web Store.

Column	Data Type	Description
WSLogID	Int	Primary Key. Web store log ID, identity column
DateTime	Datetime	Date & time of log
LogLevel	Varchar(200)	Logging level
ASPNetSessionID	Varchar(200)	Session ID
CallSite	Varchar(200)	Call site
Message	Text	Log message
StackTrace	Text	Stack trace
RemoteAddress	Varchar(500)	Remote address
HttpReferer	Varchar(500)	A misspelling of "referrer" which somehow made it into the HTTP standard. A given web page's referer (sic) is the URL of whatever web page contains the link that the user followed to the current page.
HttpUserAgent	Varchar(500)	HTTP user agent
WebServerIP	Varchar(50)	Web server IP address

### Indexes

Name	Kind	Columns	Purpose
PKWSLogID	P	WSLogID	Primary Key.

## 22.24 WSLogConfig

The WSLogConfig table is used to store logging level configuration options for the web store applications.

Column	Type	Allow Nulls	Description
WSLogConfigID	Int	N	Primary Key. Web store log ID, identity column
Application	NVarChar(50)	N	Describes the application for which the LogLevel applies. Valid options are: Consumer, PubOrderPickup and Reseller.
LogLevel	NVarChar(12)	N	Describes the LogLevel. Valid options are: Debug, Info, and Warn.
Enabled	Bit	N	Indicates whether or not the Log Level for the Application will be logged to the database. Default value is 1(True).

### Indexes

Name	Kind	Columns	Purpose
PKWSLogConfigID	P	WSLogConfigID	Primary Key.

## 22.25 WSOrderPaymentsLog

The WSOrderPaymentsLog table contains transaction results from a payment provider.

Column	Type	Allow Nulls	Description
WSOrderPaymentsLogID	integer	N	Primary Key
OrderID	integer	Y	Order associated with log record
TransactionRecord	nvarchar(max)	N	XML record
TransactionResult	Varchar(50)	N	Result of error, "Authorized", "Error", "Error", etc.

### Indexes

Name	Kind	Columns	Purpose
PKWSOrderPaymentsLogID	PK	WSOrderPaymentsLogID	Table primary key
IXWSOrderPaymentsLogOrderID	IX	OrderID	Improve querying on OrderID

## 22.26 WSPaymentBootstrapDetails

The WSPaymentBootstrapDetails table contains the information necessary to bootstrap the Web Store for an existing order. Used when a payment provider does not accept dynamic URLs for success/cancel/failure return calls.

### Columns

Column	Data Type	Allow Nulls	Description
PaymentBootstrapDetailID	Int	N	Primary Key
PaymentBootstrapHeaderID	Int	N	FK to WSPaymentBootstrapHeaders.PaymentBootstrapHeaderID
Key	nvarchar(64)	Y	Key in Key Value Pair, used to store any information needed to bootstrap the store
Value	nvarchar(64)	Y	Value in Key Value Pair, used to store any information needed to bootstrap the store

### Indexes

Name	Kind	Columns	Purpose
PKPaymentBootstrapDetailID	PK	PaymentBootstrapDetailID	Table primary key

## 22.27 WSPaymentBootstrapHeaders

The WSPaymentBootstrapHeaders table contains information necessary to connect a transaction with a payment provider to an order in the Web Store. Used with WSPaymentBootstrapDetails which holds the order information to bootstrap the Web Store. Used when a payment provider does not accept dynamic URLs for success/cancel/failure return calls.

### Columns

Column	Data Type	Allow Nulls	Description
PaymentBootstrapHeaderID	Int	N	Primary Key
PaymentProviderID	nvarchar(64)	N	Used to store the identifier provided by the Payment Provider for the transaction
OrderID	Int	N	FK to Orders.OrderID
Token	nvarchar(64)	Y	Used to store a token (if available) provided by the Payment Provider to perform future transactions without needing Credit Card Information
Status	Int	N	The current status of the transaction <sup>1</sup>

### Indexes

Name	Kind	Columns	Purpose
PKPaymentBootstrapHeaderID	PK	PaymentBootstrapHeaderID	Table primary key

### <sup>1</sup> Status Values

Value	Gateway Constant Name	Description
1	New	Transaction has been created, but no details have been saved
2	InProcess	Waiting for response from payment provider
3	Canceled	Transaction was canceled by guest
4	Rejected	Transaction was rejected. Detailed information located in WSPaymentLog.TransactionRecord errorMessage
5	Complete	Transaction has been completed successfully

## 22.28 WSPaymentConfig

The WSPaymentConfig table contains the configuration settings for payment providers. NOTE: This table has been replaced by the WSPaymentConfigHeaders and WSPaymentConfigDetails tables.

Column	Data Type	Description
WSPaymentConfigID	Int	Web store payment provider configuration ID
ConfigID	Int	Configuration ID
Key	Varchar(200)	Web store payment provider configuration key
Value	Varchar(200)	Web store payment provider configuration value

### Indexes

Name	Kind	Columns	Purpose
PKWSPaymentConfigID	PK	WSPaymentConfigID	Table primary key

## 22.29 WSPaymentConfigDetails

The WSPaymentConfigDetails table contains web admin settings for each payment provider.

Column	Data Type	Allow Nulls	Description
WSPaymentConfigDetailID	Int	N	Primary Key
WSPaymentConfigHeaderID	Int	N	Foreign Key into WSPaymentConfigHeaders table
Key	NVarChar(200)	N	Name of admin setting
Value	NVarChar(200)	N	Value for admin setting

### Indexes

Name	Kind	Columns	Purpose
PKWSPaymentConfigDetailID	PK	WSPaymentConfigDetailID	Table primary key

## 22.30 WSPaymentConfigHeaderOverrides

The WSPaymentConfigHeaderOverrides table links a Payment Provider Configuration to a Specific Category Group within the eGalaxy Consumer WebStore. This is not supported in Reseller.

Column	Data Type	Allow Nulls	Description
WSPaymentConfigHeadOverrideID	Int	N	Primary Key, always unique.
WSPaymentConfigHeaderID	Int	N	Link to WsPaymentConfigHeader table that contains the Payment Provider Information
CategoryGroupExternalId	NVarChar(50)	N	The ExternalID from SalesChannelDetails for the Category Group that is overriding the default Payment Provider Information

### Indexes

Name	Kind	Columns	Purpose
PKWSPaymentConfigHeaderOverrideID	PK	WSPaymentConfigHeaderOverrideID	Table primary key

## 22.31 WSPaymentConfigHeaders

The WSPaymentConfigHeaders table contains header records for a single payment provider.

Column	Data Type	Allow Nulls	Description
WSPaymentConfigHeaderID	Int	N	Primary Key
WSConfigHeaderID	Int	N	Foreign Key into WSConfigHeaders table
ProviderType	NVarChar(50)	N	Payment Provider type/name
Active	Bit	N	Determines if payment provider is displayed on the web store
Default	Bit	N	Provider to be selected when first presented to user
Sequence	Int	N	Used for ordering payment provider list
StoreProviderOverride	Bit	Y	Indicator stating whether the PaymentConfigHeader record contains the Payment Provider information applicable to all Category Groups, or if the record is for a Category Group Override. A True/1 value means the Header Record holds Payment Provider Configuration linked to an Category Group Override. A False/0/NULL value means the Header Record holds the Stores default Payment Provider Configuration for that Provider.
Name	NVarChar(50)	Y	Custom name for the Payment Provider Configuration

### Indexes

Name	Kind	Columns	Purpose
PKWSPaymentConfigHeaderID	PK	WSPaymentConfigHeaderID	Table primary key

## 22.32 WSPaymentThemeDetails

The WSPaymentThemeDetails table will store identifiers based on the payment provider and web store theme names. These IDs are used for theming indirect payment provider's websites.

### Columns

Column	Data Type	Allow Nulls	Description
WSPaymentThemeDetailID	Int	N	Primary Key
WSPaymentConfigHeaderID	Int	N	Foreign Key to WSPaymentConfigHeader.WSPaymentConfigHeaderID
ThemeName	nvarchar(64)	N	Reference to an existing theme that is set up in Merchants.WebTheme
CssId	nvarchar(64)	N	Identifier required by Payment Provider to apply a theme to their checkout page.

### Indexes

Name	Kind	Columns	Purpose
PKWSPaymentThemeDetailID	PK	WSPaymentThemeDetailID	Table primary key

### 22.33 WSPlugins

The WSPlugins table consists of one or more installed plugins.

Column	Data Type	Allows Nulls	Description
WSPluginID	Int	N	Primary key, always unique
PluginName	NVarchar(100)	N	Plugin Name
PluginDescription	text	Y	Description of the plugin
PluginGUID	Unique Identifier	N	A Unique Identifier that is used to identify and instantiate the plugin
Version	NVarChar(10)	N	Plugin Description
PluginType	NVarChar(50)	N	Type of Plugin
IsActive	Bit	N	Bit value to indicate if the plugin is active mode or not

#### Indexes

Name	Kind	Columns	Purpose
PKWSPlugins	P	WSPluginID	Primary Key

## 22.34 WSPublishingLog

The WSPublishingLog table is used by the eGalaxy.ashx publishing handler to log information during publish or order pickup.

Column	Data Type	Description
WSPublishingLogID	Int	Primary key, identity column
DateTime	DateTime	Date/time of the log entry
LogLevel	VarChar(200)	Level of the log message: Debug, Info, Warn, Error
ASPNetSessionID	VarChar(200)	ASP.NET generated session id for the user
CallSite	VarChar(200)	Code location from which the log event occurred
Message	Text	Free form text
StackTrace	Text	Full .NET stack trace that is recorded for errors
RemoteAddress	VarChar(500)	User's IP address
HttpReferer	VarChar(500)	Referring page for the current page request
HttpUserAgent	VarChar(500)	Web browser identification string
WebServerIP	Varchar(50)	Web server IP address

### Indexes

Name	Kind	Columns	Purpose
PKWSPublishingLogID	P	WSPublishingLogID	Primary Key.

## 22.35 WSPublishStatus

The WSPublishStatus table stores and tracks publish information in the web store.

Column	Data Type	Allow Null	Description
WSPublishStatusID	Int	N	Primary key, always unique
SalesChannelPublishCommandID	Int	N	ID sent by WebPublishing to reference back to Galaxy
CreateDate	DateTime	N	Timestamp the publish request was made
PublishStatusID	Int	N	Numeric Status of the publish request <sup>1</sup>
PublishStatusText	NVarChar(max)	N	Text Description for the PublishStatusID
CompleteDate	DateTime	Y	Timestamp the publish Completed - Failed/Succeeded

### Indexes

Name	Kind	Columns	Purpose
PKWSPublishStatusID	P	WSPublishStatusID	Primary Key

<sup>1</sup> PublishStatusID Values

Value	Description
0	Unknown
1	In Process
2	Fail
3	Success

## 22.36 WSSecurityIPAddresses

The WSSecurityIPAddresses table is used to store the IP addresses that the online administration panel is restricted to responding to.

Column	Data Type	Allow Null	Description
WSSecurityAddressesID	Int	N	Web store security addresses ID
IPAddress	Varchar(50)	N	IP address
AdminPanel	Bit	Y	Determines if IP Address will be restricted for the Admin Panel on the web store. Defaults to True.
Publishing	Bit	Y	Determines if IP Address will be restricted for Publishing/Order Pickup.
Orders	Bit	Y	Determines if IP Address will be restricted for Order Pickup, Status changes, etc.

### Indexes

Name	Kind	Columns	Purpose
PKWSSecurityIPAddressID	PK	WSSecurityIPAddressID	Table Primary Key

## 23 Miscellaneous

The tables listed in this section are shared tables between the modules.

### 23.1 ApiUsage

This table contains the usage logs for various API calls throughout the system.

#### Columns

Column	Type	Allow Nulls	Description
ApiUsageID	BigInt	N	Primary key, always unique.
Application	NVarChar(50)	N	The name of the Application is providing the API.
CorrelationID	UniqueIdentifier	N	GUID that links request and response records together.
Timestamp	DateTime	N	The date and time that the request/response was processed.
UsageType	Integer	N	Type of Usage <sup>1</sup> .
Content	NVarChar(MAX)	N	The contents of the request or response.
ContentLength	Integer	N	The length in bytes of the content.
Headers	NVarChar(2000)	Y	The HTTP headers included in the request/response.
Uri	NVarChar(500)	Y	The Universal Resource Identifier of the request/response.
IPAddress	NVarChar(100)	Y	The IP Address where the request/response originated.
User	NVarChar(100)	Y	The authenticated user information that is connected to this request/response.
RequestMethod	NVarChar(250)	Y	The HTTP request method.
ControllerName	NVarChar(250)	Y	The name of the internal controller responsible for handling the request.
Action	NVarChar(100)	Y	The name of action that the controller performed to satisfy the request.
StatusCode	Integer	Y	The HTTP status code of the response <sup>2</sup> . This will be zero for requests.
ApiKey	NVarChar(150)	Y	The API Key included with the request.

#### Indexes

Name	Kind	Columns	Purpose
PKApiUsageID	P	ApiUsageID	Primary Key.
IXApiUsageCorrelationID	I	CorrelationID	To facilitate linking of request and response records by CorrelationID

<sup>1</sup> UsageType Values

Value	Description
0	Request
1	Response

<sup>2</sup> StatusCode Values (See <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html> for detailed descriptions)

Values	Description
0	None - Used for request entries only.
200 - 299	Success
300 - 399	Redirection
400 - 499	Client Error
500 - 599	Server Error

## 23.2 Countries

This table contains the Country name information to be used in MWS, Order Entry, Passes and the eGalaxy Web Store.

### Columns

Column	Type	Allow Nulls	Description
CountryId	Int	N	Primary key, always unique.
CountryCode	Char(2)	N	Alpha-2 code based on the ISO Country List. Alternate Key, always unique
ShortName	Varchar(80)	Y	Country short name
OfficialName	Varchar(80)	Y	Country Official name
Alpha3Code	Char(3)	Y	Alpha-3 Country code.
Numeric3Code	Char(3)	Y	Numeric-3 code
Inactive	Bit	Y	Determines if the country is Inactive for eGalaxy Web Store.
CountryGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

### Indexes

Name	Kind	Columns	Purpose
PKCountryId	P	CountryId	Primary Key.
AKCountryCode	A	CountryCode	Alternate Key

### 23.3 GatewayCounters

This table provides a unique ID when inserting a new row into a table supporting Business Objects. Each Primary Key column in the database must have a unique name in order for this to work. For example, two tables cannot have a Primary Key column named "ID".

#### Columns

Column	Type	Allow Nulls	Description
Name	VarChar(30)	N	Name of table that needs a unique incremental number. This column is also the Primary Key for this table.
CountValue	Int	N	Current value of Counter Name. Highest unique ID number assigned to the table labeled as the Counter Name.

#### Indexes

Name	Kind	Columns	Purpose
PKGatewayCountersName	P	Name	Primary Key.

### 23.4 GatewayLocks

Each record within the GatewayLocks table represents a single lock for a table or a specific row within a table. Each record contains what is locked, who has locked it and when it was locked. There is no business object associated with this table; therefore it does not have our standard UniqueID column. This table will only be used when locking has been enabled on a PersistenceManager.

#### Columns

Column	Type	Allow Nulls	Description
TableName	Char(30)	N	Name of the table that is locked, or the table containing the row that is locked.
ApplicationName	Char(30)	Y	Name of the application that owns this lock record.
RowID	Int	N	UniqueID of the row being locked. If the lock record is for a table, this value will be zero.
ProcessID	Int	N	Numeric ID of the thread that created the lock record. This is checked because multi-threaded applications will have the same node number and machine name.
NodeNumber	Int	Y	Node number of the machine locking the table/row.
MachineName	Char(100)	Y	Machine name of the machine locking the table/row, used when no node number is available.
LockType	Int	N	Type of lock <sup>1</sup>
LockDateTime	DateTime	N	Date and Time the lock was added

#### Indexes

Name	Kind	Columns	Purpose
PKGatewayLocksTableNameRowID	P	TableName, RowID	Primary key.

<sup>1</sup> LockType Values

Value	Description
0	Row
1	Table

## 23.5 GXEventLog

This table contains events that occur in Galaxy and its modules.

### Indexes and Constraints

Column	Type	Allow Nulls	Description
GxEventLogUID	Int	N	Primary key, always unique
EventDateTime	DateTime	N	Date and time of the event
NodeID	Int	Y	Node number where the event occurred
UserID	Int	Y	User (Agent) ID that was logged on when the event occurred
Severity	Int	N	Severity of the event <sup>1</sup>
ApplicationName	Char(40)	N	Name of Application (without .exe extension) that invoked the event
ProcessName	Char(40)	Y	Name of the process within the Application that invoked the event
EventText	Char(1000)	Y	Description of the event
EventData	Char(100)	Y	Any data associated with this event
ErrorCode	Int	Y	Any error code associated with this event
EventCode	Int	Y	A unique code corresponding to each type of event <sup>2</sup>
EventSubCode	Int	Y	A unique sub-code that when combined with the EventCode yields a unique identifier for the event type <sup>3</sup>
AgencyNo	Int	Y	Agency number where the event occurred. Foreign key to Agencies.AgencyNo.

### Indexes

Name	Kind	Columns	Purpose
PKGxEventLog	P	GxEventLogUID	Primary key
IXGxEventLogEventDateTime		EventDateTime	Improve queries including date/time
IXGxEventLogViewLog		Severity, ApplicationName, ProcessName, EventDateTime	Improve report performance

### <sup>1</sup> Severity Values

Value	Description
0	Information
1	Warning
2	Error

### <sup>2</sup> EventCode Values

Value	Gateway Constant Name	Description
0	ecNone	No event code defined for this event. EventSubCode is always zero for this value.
1	ecMWSExecuteACSFunction	MWS executed an ACS function.
2	ecJournalSenderError	A journal sender error. EventSubCode is always zero for this value.
3	ecPCIEvent	An event affecting system security (as defined by PCI-DSS v2.0).
4	ecConfigurationChange	
5	ecSupervisorApproval	Supervisor approval event. The event subcode for this event code is the SAReason code value (Defined for the JnlHeaders table)
6	ecCentralDatabase	Central Database event. EventSubCode will always be assigned for this value.
7	ecHTMLKiosk	HTML Kiosk event. EventSubCode will always be assigned for this value.
8	ecSalesSecurity	Sales security events. An EventSubCode will be assigned with these types based on the sales event that was recorded.
9	ecPrintingFiscalReports	Printing fiscal reports. An EventSubCode will be assigned with these types based on the report that was printed.
10	ecRemoveLock	Remove GatewayLock for ticket/pass. EventSubCode will always be assigned for this value.
12	ecContactMerge	One or more contacts merged into a single contact. EventSubCode is always zero for this value.

### <sup>3</sup> EventSubCode Values for Event Code ecMWSExecuteACSFunction. These correspond directly to values stored in FKeyTasks.Task.

Value	Gateway Constant Name	Descriptions
0	EXIT_MENU_TASK	ACS32 exit menu task
1	LOGOFF_TASK	Log current agent off of terminal.
2	SCAN_TASK	Enter a pre-defined scan, or allow entry from keyboard.
3	PAGE_TASK	Page Tlead station.
4	ADMIT_TASK	Admit ticket holder (record usage) if last scan was invalid.
5	HELP_TASK	Display help message.
6	TIME_TASK	Display current date/time.
7	MESSAGES_TASK	View messages from Tlead.
8	DATE_TASK	Set the date/time stamp for scanning.
9	RELOCK_TASK	Relock turnstile.
10	MENU_TASK	Switch to another menu.
11	EXIT_MODE_TASK	Turnstile: Allow exit only. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
12	ENTRY_MODE_TASK	Turnstile: Allow enter only. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
13	PAID_MODE_TASK	Turnstile: Allow PAID enter and exit. This function is obsolete: it has been replaced by SET_ACP_MODE_TASK.
14	FREE_MODE_TASK	Turnstile: Allow FREE entry/exit. This function is obsolete: it has been replaced by ACP_MODE_TASK.
15	SHOW_MODE_TASK	Turnstile: Display the current ACP mode last set by invoking the ACP_MODE_TASK.
16	GETVER_TASK	Turnstile: Display version.

17	RESTART_TASK	Turnstile: Restart
18	RECORD_TASK	Turnstile: Record Entry.
19	TEMPCLOSE_TASK	Turnstile: Close temporarily.
20	UNLOCK_TASK	Turnstile: Unlock mode.
21	GROUP_EXIT_TASK	Group Exit.
22	CLOSE_TASK	Close station.
23	LOCKBAR_TASK	Turnstile: Lock turnbar once.
24	UNLOCKBAR_TASK	Turnstile: Unlock turnbar once.
25	REPLACE_OPER_TASK	Replace current operation.
26	CLEAR_OPER_TASK	Clear all active operations.
27	ADD_OPER_TASK	Add a single operation.
28	REMOVE_OPER_TASK	Remove a single operation.
29	SHOW_OPER_TASK	Show active operations.
30	ACP_MODE_TASK	Set ACP Mode, or show menu
31	MANAGE_DEVICES_TASK	Manage devices in ACS32
32	EXIT_TASK	Exit application in ACS32
33	MANAGE_MODULES_TASK	Manage AX1180 modules in ACS32
34	AUTOSCAN_TASK	AutoScanning in ACS32
35	SHOW_CONSOLE_TASK	Show console in ACS32
36	TICKET_LOOKUP_TASK	Ticket lookup in ACS32
37	NEXT_TICKET_LOOKUP_TASK	Lookup next ticket scanned in ACS32
38	START_SELF_SERVICE_TASK	Start ACP Self-Service Mode
39	STOP_SELF_SERVICE_TASK	Stop ACP Self-Service Mode
40	TOGGLE_BIOMETRICS_TASK	Stop using biometrics until restart
41	TOGGLE_LOGGING_TASK	Toggle all logging
42	EXIT_GALAXY_TASK	Exit the Galaxy Point of Sale application (used only when ACS32 is running within Galaxy)
43	CAPTURE_BIOMETRIC_TASK	Performs Registration of Biometric data
44	RELOCK_ALL_TASK	Relock the turnbar for all pending guests. This function removes all pending rotations on the turnbar.
45	SHOW_BIOMETRICS_STATE_TASK	Show whether biometrics are currently enabled or disabled.
46	ENABLE_BIOMETRICS_TASK	Enable biometric activities.
47	DISABLE_BIOMETRICS_TASK	Disable biometric activities.
48	SHOW_BIOMETRICS_RATE_TASK	Show the current rate at which biometric identifications are being performed.
49	SET_BIOMETRICS_RATE_TASK	Set the rate at which to perform biometric identifications.
50	SHOW_LOGGING_TASK	Show whether the logging function has been invoked to enable diagnostic logging.
51	ENABLE_LOGGING_TASK	Enable all diagnostic logging. All logging categories are enabled and ACS32 logs everything directly to the log file. (Logging is automatically disabled when a user logs off or exits ACS32).
52	DISABLE_LOGGING_TASK	Disable diagnostic logging. All log categories are turned off.
53	EXECUTE_GALAXY_FUNCTION_TASK	Execute galaxy function in ACS@POS
54	CHANGE_ACP_TASK	Change the current ACP
55	ENABLE_VEHICLE_DETECTOR_TASK	Enable vehicle detector
56	DISABLE_VEHICLE_DETECTOR_TASK	Disable vehicle detector
57	TOGGLE_VEHICLE_DETECTOR_TASK	Toggle vehicle detector
58	TOGGLE_BYPASS_VEHICLE_DETECTOR_TASK	Toggle bypass vehicle detector
59	REINITIALIZE_VEHICLE_DETECTOR_TASK	Reinitialize vehicle detector

<sup>3</sup> EventSubCode Values for Event Code ecPCIEvent .

Value	Gateway Constant Name	Descriptions
1	ENABLE_USER_PRIVILEGE_EVENT	Enable privilege
2	DISABLE_USER_PRIVILEGE_EVENT	Disable privilege
3	UNSUCCESSFUL_LOGIN_ATTEMPT	Failed login attempt
4	NEW_USER_PROFILE_CREATED	New user profile created
5	PASSWORD_RESET	Account unlocked
6	PASSWORD_CHANGE	Password change
7	USERS_PRIVILEGE_PROFILE_CHANGED	User's profile changed
8	UPDATE_ENCRYPTION_KEY	Encryption key changed
9	REMAINING_LOGINTRIES_CHANGED	Increase remaining login attempts
10	CREATE_NEW_USER_PRIVILEGE_PROFILE	New privilege profile created
11	VIEW_UNENCRYPTED_PAYMENT_CARD_NUMBER	View unencrypted payment card number
12	CENTRAL_DATABASE_PUBLISH	Central database publish
13	VIEW_EVENT_LOG	View event log
14	SUCCESSFUL_LOGON	Successful login
15	EXPORT_EVENT_LOG	Export event log
16	CONFIGURATION_OPTION_CHANGED	Configuration option changed
17	CONFIGURATION_OPTION_ADDED	Configuration option added
18	CONFIGURATION_OPTION_REMOVED	Configuration option removed
19	INACTIVITY_LOCK_ACTIVATED	Inactivity lock screen activated
20	PROTOCOL_RECORD_UPDATES	Protocol updated
21	APPLICATION_STARTUP	Application startup
22	APPLICATION_SHUTDOWN	Application shutdown

23	TERMINAL_ID_UPDATES	Terminal ID updated
24	FOP_UPDATES	FOP updated
25	CARD_RANGE_UPDATES	Card Range updated
26	NODE_UPDATES	Node updated
27	NETWORK_CONFIGURATION_CHANGED	Network Configuration changed
28	SYSTEM_CONFIGURATION_CHANGED	System Configuration changed
29	VISANET_TERMINAL_ID_UPDATES	VisaNet Terminal ID updated
30	ACCOUNT_LOCKED	
31	NEW_USER_PROFILE_NOT_CREATED	
32	WEBSTORE_PUBLISH_START	
33	WEBSTORE_PUBLISH_SUCCESS_END	
34	WEBSTORE_PUBLISH_FAILURE_END	
35	NODEDATA_REFRESH_NODE	
36	NODEDATA_DOWNLOAD_FAIL	
37	PURGE_PUBLISH_HISTORY	
38	HTMLKIOSK_EVENT	
39	TRANSACTION_CANCEL_EVENT	
40	ITEM_REDUCTION_EVENT	
41	NO_SALE_EVENT	
42	AGENCY_REPORT_EVENT	
43	SHIFT_REPORT_EVENT	
44	DEFERRED_ENTITLEMENT_ADD_ON_ADDED	
45	GUEST_PHOTO_DELETED_EVENT	
46	MODIFIED_UPGRADE_VALUE_EVENT	
47	REMOVED_GATEWAYLOCK_EVENT	
48	REFUND_IN_DOWNGRADE_ALLOWED_EVENT	Supervisor approved refund during downgrade
49	PRICE_NOT_HONORED	
50	BLOCKED_SALES_EVENT	A sales action has been blocked, for instance if a lockout has been detected on a product

<sup>3</sup> EventSubCode Values for Event Code ecConfigurationChange .

Value	Gateway Constant Name	Descriptions
28	SYSTEM_CONFIGURATION_CHANGED	System Configuration changed

<sup>3</sup> EventSubCode Values for Event Code ecCentralDatabase .

Value	Gateway Constant Name	Descriptions
35	NODEDATA_REFRESH_NODE	Database Refresh Initiated on another Node
36	NODEDATA_DOWNLOAD_FAIL	Database Download Failure
37	PURGE_PUBLISH_HISTORY	A purge of the publish history

<sup>3</sup> EventSubCode Values for Event Code ecHTMLKiosk .

Value	Gateway Constant Name	Descriptions
38	HTMLKIOSK_EVENT	General HTML Kiosk Log Event

<sup>3</sup> EventSubCode Values for Event Code ecSalesSecurity .

Value	Gateway Constant Name	Descriptions
39	TRANSACTION_CANCEL_EVENT	A transaction was canceled.
40	ITEM_REDUCTION_EVENT	A quantity was reduced in a transaction or order.
41	NO_SALE_EVENT	The 'No Sale' function was executed.
42	AGENCY_REPORT_EVENT	
43	SHIFT_REPORT_EVENT	
44	DEFERRED_ENTITLEMENT_ADD_ON_ADDED	
45	GUEST_PHOTO_DELETED_EVENT	Guest photo was deleted from a ticket in ticket lookup.
46	MODIFIED_UPGRADE_VALUE_EVENT	An upgrade value was changed in a transaction or an order.
47	REMOVED_GATEWAYLOCK_EVENT	
48	REFUND_IN_DOWNGRADE_ALLOWED_EVENT	Return of funds authorized during downgrade

<sup>3</sup> EventSubCode Values for Event Code ecPrintingFiscalReports .

Value	Gateway Constant Name	Descriptions
42	AGENCY_REPORT_EVENT	An agency report was printed.
43	SHIFT_REPORT_EVENT	A shift report was printed.

<sup>3</sup> EventSubCode Values for Event Code ecRemoveLock .

Value	Gateway Constant Name	Descriptions
47	REMOVED_GATEWAYLOCK_EVENT	A GatewayLock was removed for a ticket/pass.

## Obtaining information from the GxEventLog table

Here are some example queries that show how to obtain information about specific events recorded in GxEventLog:

```
-- View the result of all actions where MWS executed the SET BIOMETRICS RATE TASK
SELECT EventDateTime, UserID, EventData AS ACP, ErrorCode, EventText AS Result
FROM GXEventLog WITH (NOLOCK)
WHERE EventCode = 1 /* ecMWSExecuteACSFuction */ AND EventSubCode = 49 /* SET_BIOMETRICS_RATE_TASK */
```

EventDateTime	User	ACP	ErrorCode	Result
2010-08-13 11:34:09.000	99	1	0	Rate set to 4
2010-08-13 11:36:49.000	99	1	1	Biometrics not enabled (5)
2010-08-13 12:40:53.000	99	1	2	6

For the first row, the user set the biometrics rate to 4 and the operation was successful. Note that the result of the operation is stored in the **EventText** column. The ACP affected by the operation is stored in the **EventData** column.

In the second row, the user attempted to set the biometrics rate to 5 but the response from the ACP was "Biometrics not enabled". The **ErrorCode** indicates that a general error occurred. The **EventText** column indicates the error message; the original data used for the ACS function is appended to the error message in parenthesis.

In the third row the user attempted to set the biometrics rate to 6, but the ACP never responded to MWS. The ACP may have been offline, or may not have received the message, or MWS may not have received a response from the ACP. The data provided with the function is found in the **EventText** column.

```
-- View all actions where MWS executed an ACS function and a response was never received from the ACP
SELECT EventDateTime, UserID, EventCode, EventSubCode, EventData AS ACP, EventText AS FunctionData
FROM GXEventLog WITH (NOLOCK)
WHERE EventCode = 1 /* ecMWSExecuteACSFuction */ AND ErrorCode = 2 /* ecTimeoutError */
```

EventDateTime	User	EventCode	EventSubCode	ACP	FunctionData
2010-08-13 11:36:49.000	99	1	51	5	

In the first row the user attempted to execute the ENABLE\_LOGGING\_TASK function, but the ACP never responded to MWS. The ACP may have been offline, or may not have received the message, or MWS may not have received a response from the ACP. The data provided with the function is found in the **EventText** column. **EventText** was renamed as **FunctionData** for clarity. Since there is no function data for the ENABLE\_LOGGING\_TASK none is displayed here.

```
-- View all actions where MWS executed an ACS function and the ACP failed to execute the function for an unknown reason
SELECT EventDateTime, UserID, EventCode, EventSubCode, EventData AS ACP, EventText AS ErrorMessage
FROM GXEventLog WITH (NOLOCK)
WHERE EventCode = 1 /* ecMWSExecuteACSFuction */ AND ErrorCode = 1 /* ecGeneralError */
```

EventDateTime	User	EventCode	EventSubCode	ACP	ErrorMessage
2010-08-13 11:36:49.000	99	1	49	3	Biometrics not enabled (2)

In the first row the user attempted to execute the SET\_BIOMETRICS\_RATE\_TASK function, but the ACP responded that Biometrics was not enabled at the ACP. The data provided with the function is found in the parenthesis in the **EventText** column. **EventText** was renamed as **ErrorMessage** for clarity.

### 23.6 GXExchangeRates

This table contains foreign currency information.

#### Indexes and Constraints

Primary Key: PKExchangeRateID

Column	Type	Allow Nulls	Description
ExchangeRateID	Int	N	Primary key, always unique
Name	Char(2)	N	The single-letter currency key ('A' thru 'Z' inclusive)
Description	Char(8)	N	The currency description
Rate	Float	N	The actual exchange rate of the currency
Abbreviation	char(3)	Y	The abbreviation for this currency
Decimals	Int	Y	The number of decimal places in the Rate
Rounding	Int	Y	

### 23.7 GXFileConnections

The GxFileConnections table contains links rows in the GxFiles table to different entities based on the ConnectionType column.

#### Columns

Column	Type	Allow Nulls	Description
GxFileConnectionID	Int	N	Primary key, always unique
GxFileID	Int	N	Foreign key to GxFiles.GxFileID, the file that this connection points to
ConnectionType	Int	N	The type of this file connection <sup>1</sup>
ConnectionID	Int	N	The UniqueID of the row this file is connected to, table is based on the ConnectionType

#### Indexes

Name	Kind	Columns	Purpose
PKGxFileConnectionID	P	GxFileConnectionID	Primary Key.
IXConnectionTypeConnectionID	A	ConnectionType, ConnectionID	Used when querying by type and ID

<sup>1</sup> ConnectionType Values

Value	Gateway Constant Name	Description
1	ID_EVENT_TYPE_CONNECTION	Connection points to the RMEventTypes table
2	ID_EVENT_CONNECTION	Connection points to the RMEvents table

### 23.8 GXFiles

The GxFiles table contains files that can be attached to an Event Type or Event. These files will then be attached to any confirmation e-mail for PrintAtHome tickets utilizing the Event.

#### Columns

Column	Type	Allow Nulls	Description
GxFileID	Int	N	Primary key, always unique
Description	VarChar(255)	N	Description for this file
Filename	VarChar(128)	N	Filename of this file
Filesize	Int	N	Size of the file in bytes
FileData	Image	N	Binary data for this file

#### Indexes

Name	Kind	Columns	Purpose
PKGxFileID	P	GxFileID	Primary Key.

### 23.9 GXProjections

This table contains attendance projections.

#### Indexes and Constraints

Primary Key: PKProjectionID

Indexes:

(None)

Column	Type	Allow Nulls	Description
ProjectionID	Int	N	Primary key, always unique
ProjectionDate	DateTime	N	The projected attendance date
Attendance	Int	N	The total number of arrivals
Promotion	char(60)	Y	
Weather	char(60)	Y	
Other	char(60)	Y	

### 23.10 GxRecordLog

The table stores the changes of **Users**, **Agencies**, **User Profiles**, and **Item Groups** from Galaxy.

#### Columns

Column	Type	Allow Nulls	Description
GxRecordLogID	Int	N	Primary key, always unique
UserID	Char(60)	N	User name (User ID)
NodeID	Int	N	Node the entry was made from
ModifyDate	DateTime	N	Date and Time the modification was made
Action	Int	N	Type of action for this entry <sup>1</sup>
TableID	Int	N	Constant ID which points to a particular table. See the Table IDs section for a full list of possible values.
RecordID	Char(30)	N	The unique identifier for the record. For most SQL tables, this value actually translates to the integer Unique ID.
RecordDescription	Char(60)	N	A character description of the entry
OldData	VarChar(3750)	Y	The data the field used to contain
NewData	VarChar(3750)	Y	The data the field now contains
FieldName	Char(40)	Y	The name of the field that changed
GroupID	Int	Y	The group ID for this entry, to link entries together as being part of one "change"
ApplicationID	Int	Y	ID referencing the application that created this entry <sup>2</sup>
GxKeyID	Int	Y	The encryption key that was used to encrypt the OldData and NewData fields for the current record

#### Indexes

Name	Kind	Columns	Purpose
PKGxRecordLogGxRecordLog	PK	GxRecordLogID	Primary Key
IXGxRecordLogTableIDRecordID		TableID, RecordID	Index to speed up searching.
IXGxRecordLogGroupID		GroupID	Index to speed up searching.
IXGxRecordLogModifyDate		ModifyDate	Index to speed up searching.

<sup>1</sup> Action Values

Value	Const Name	Descriptions
1	laInsert	Record was inserted
2	laDelete	Record was deleted
3	laModify	Record was modified

<sup>2</sup> ApplicationID Values

Value	Const Name	Descriptions
0	LEGACY_APPLICATION_ID	Legacy records will default this field to null, and will be read as zero in the system.
1	GALAXY_APPLICATION_ID	Record was created by Galaxy
2	eGALAXY_APPLICATION_ID	Record was created by eGalaxy
3	MANAGERS_WORKSTATION_APPLICATION_ID	Record was created by Manager's Workstation
4	WEB_ORDER_PROCESSOR_APPLICATION_ID	Record was created by Web Order Processor

### 23.11 GxTriggers

This table contains entries to indicate the Gateway utility application to extract the indicated data from SQL table and export them to external media per ProcessCode.

#### Columns

Column	Type	Allow Nulls	Description
GxTriggerUID	Int	N	Primary key, always unique, Identity
TableName	Varchar(128)	N	Table name for this entry
RecordID	Int	N	Unique Key value for TableName entry
ProcessCode <sup>1</sup>	Int	N	Used by the External extract application to tell how to process this entry
Action <sup>2</sup>	Int	N	
NodeNo	Int	Y	Node number who created this entry
ProcessStatus <sup>3</sup>	Int	N	Current status of this entry
CreateDate	DateTime	N	Creating date of this entry
Tries	Int	Y	The number of attempts that have been made to process this trigger record. This column is only used by some processors, so may have a zero value in some instances.
Priority <sup>4</sup>	Int	Y	Indicates the priority in which the trigger will be processed. This value may not apply to all processors. A value of NULL indicates a normal priority.
LastError	Varchar(1000)	Y	This column will contain the last known error if this given trigger is in an error state
GalaxySiteID	Int	Y	SiteID of an attraction as defined by Gateway Ticketing Systems, Inc.

#### Indexes

Name	Kind	Columns	Purpose
PKGxTriggerID	P	GxTriggerUID	Primary Key.
IXGxTriggersTblCdeNdeSttsPrty	A	TableName, ProcessCode, NodeNo, ProcessStatus, Priority	Index used for loading triggers in REConsole
IXGxTriggersPriority index	A	Priority	Index used for speeding up queries that order GxTriggers by priority.

#### <sup>1</sup> ProcessCode Values

Value	Gateway Constant Name	Description
1	PC_MAGICQUEST	Record is for the Magic Quest Process (MServer)
2	PC_HANDSHAKE	Record is for the Handshake Process (HSServer)
3	PC_REMERGE	Record is for the REMerge Process
4	PC_SWSERVER	Record is for the SeaWorld Server (SWServer)
5	PC_HCWSERVER	Record is for Hershey's Chocolate World Server (HCW Server)
6	PC_BATCH_SETTLE	Record is for VisaNet batch processing
7	PC_MULTI_SITE	Record is for the MultiSite Server application
8	PC_EXPORT_SERVICE	Record is for the Export Service Application
9	PC_POST_USAGE_SERVICE	Record is for the Post Usage Service
10	PC_SYSLOG_SENDER_SERVICE	Record is for the SysLog Sender Service
11	PC_WEB_PUBLISHING_SERVICE	Trigger is for Web Publishing service and reflects a publishing task.
12	PC_BIN_LOOKUP_SERVICE	Trigger is for Bin Lookup service.
13	PC_GALAXY_CONNECT_SERVICE	Trigger is for the Connect Proxy Service.
14	PC_SIHOST_PMS_SERVICE	Trigger is for the SIHOST PMS service.
15	PC_FIPAY_TRANSACTION_UPDATE_SERVICE	Trigger is for the FIPay Transaction Update service.
16	PC_CONNECT_PUBLISHING_SERVICE	Trigger is for the Web Publishing service and reflects a Galaxy Connect publishing task.

#### <sup>2</sup> Action Values

Value	Gateway Constant Name	Description
1	ACTION_INSERT	
2	ACTION_DELETE	
3	ACTION_UPDATE	

#### <sup>3</sup> ProcessStatus Values

Value	Gateway Constant Name	Description
0	STATUS_AVAILABLE	Record is available to be processed
1	STATUS_PROCESSING	Flagged to be processed
2	STATUS_ERROR	Error occurred processing this transaction
3	STATUS_COMPLETE	Transaction processed with no errors
4	STATUS_SKIP	This trigger will be skipped next time it comes around for processing.
5	STATUS_PROC_REJECT	Indicates the record was processed by rejected trigger thread.
6	STATUS_CLAIMED	Any service that processes multiple triggers at a time can claim a group of triggers for processing. This is an intermediate status between STATUS_AVAILABLE and STATUS_PROCESSING. If a trigger is marked with ProcessStatus STATUS_CLAIMED, it has claimed a trigger for processing but has not yet started processing the trigger.
7	STATUS_ONHOLD	Trigger started processing and is now waiting for an external operation to complete. Currently used for Web Publishing, Async mode. Next status would be STATUS_COMPLETE or STATUS_ERROR.

**4 Priority Values**

Value	Gateway Constant Name	Description
0	PRIORITY_NORMAL	Indicates a normal priority. Triggers with this value are processed before low priority triggers
1	PRIORITY_LOW	Indicates a lower than normal priority. Triggers with this value are processed only when all normal priority triggers are all processed

## 23.12 LogEntries

This table will be used to consolidate log data from Galaxy and the Galaxy Connect Service.

### Columns

Column	Type	Allow Nulls	Description
LogEntryID	BigInt	N	Primary key, always unique (Identity).
Timestamp	datetime2	N	The local date and time of the log entry.
Severity	Integer	N	The integer or ordinal value of the log entry severity, using the NLog standard <sup>1</sup> .
SeverityText	nvarchar(50)	Y	The string value of the log entry severity.
Message	nvarchar(1500)	Y	The log message.
Title	nvarchar(500)	Y	The log title (optional).
EventID	Integer	Y	The log event id (optional).
Priority	Integer	Y	The log priority value of 1-10 (optional).
MachineName	nvarchar(50)	Y	The name of the machine running the process.
AppDomainName	nvarchar(512)	Y	The name of the executable that requested the log entry.
ProcessID	nvarchar(256)	Y	The current process id related to the log entry.
ProcessName	nvarchar(512)	Y	The current process name related to the log entry.
ThreadID	nvarchar(50)	Y	The current thread id related to the log entry.
ThreadName	nvarchar(512)	Y	The current thread name related to the log entry.
ExtendedProperties	nvarchar(1000)	Y	The log entry extended information. (optional)
ActivityID	uniqueidentifier	Y	A unique identifier for the current log entry.
RelatedActivityID	uniqueidentifier	Y	A unique identifier for a related log entry.
ErrorMessages	nvarchar(max)	Y	The log error message text and/or stack trace.
Categories	nvarchar(500)	Y	A list of comma separated values that represent categories of log entries, ie. "Queue, Application"

### Indexes

Name	Kind	Columns	Purpose
PKLogEntries	P	LogEntryID	Primary Key.

### <sup>1</sup> Severity Values

Value	Description
0	Trace
1	Debug
2	Info
3	Warn
4	Error
5	Fatal
6	Off

### 23.13 MessageQueue

This table stores messages to be sent by the Galaxy Message Sender Service. The service periodically scans this table for unsent messages and sends them when found.

eGalaxy and Galaxy are applications that update this table.

#### Columns

Column	Type	Allow Nulls	Description
MessageQueueID	Integer	N	Unique ID
Kind <sup>1</sup>	Integer	N	Kind of message, see Kind values
Priority <sup>2</sup>	Integer	N	Priority of message, see Priority values <sup>2</sup> . Note that Priority does not affect the sender of the message (i.e. the Message Sender Service). Instead, the recipient acts upon the priority.
SenderName	NVarChar(64)	N	Name of sender
SenderAddress	NVarChar(64)	N	Address of sender
RecipientName	NVarChar(max)	Y	Name of recipient(s), separated by commas
RecipientAddress	NVarChar(max)	Y	Addresses of recipients, separated by commas
ReplyToAddress	NVarChar(max)	Y	Reply-to email address
Subject	NVarChar(128)	Y	Subject of message
ContentType <sup>3</sup>	Integer	Y	The format of the data stored in Body column, see ContentType values
Body	NVarChar(max)	Y	Body of message. In a multipart e-mail, this represents the plain text portion of the e-mail.
SendDate	DateTime	Y	The date the message was sent, if sent
ReleaseDate	DateTime	Y	The date the message should be sent
Status	Integer	N	Status, see status values <sup>4</sup>
ErrorText	NVarChar(max)	Y	Error description, if status = 99
HTMLBody	NVarChar(max)	Y	HTML portion of a multipart e-mail.
RetryCount	Int	Y	Number of times to retry sending an email when an error occurs.
ExternalID	NVarChar(40)	Y	External ID for this entry. For order-related e-mails, this will be the Orders.ExternalID value.
DeliveryMethodID	Int	Y	When a message's Kind is a kind that is delivered by way of an external provider, such as Kind 2 (SMS delivery), it's possible for more than one provider to be supported. The DeliveryMethodID is an internal identifier that refers to the delivery agency.
TransactionID	NVarChar(32)	Y	Contains the ID that is generated by the remote messaging service that was used to send this message.

#### Indexes

Name	Kind	Columns	Purpose
PKMessageQueueMessageQueueID	P	MessageQueueID	Primary key.
IXMessageQueueStatusReleaseDate		Status, ReleaseDate	Used by the query to check for unsent messages
IXMessageQueueStatusSendDate		Status, SendDate	Used by the query when purging messages

#### <sup>1</sup> Kind Values

Value	Gateway Constant Name	Description
1	KIND_SMTP	Email Message
2	KIND_SMS	Text Message

#### <sup>2</sup> Priority Values

Value	Gateway Constant Name	Description
0	MpHighest	The message should be given the highest priority
1	MpHigh	This message should be given a high priority
2	MpNormal	This message should be given the normal priority.
3	MpLow	This message should be given a low priority.
4	MpLowest	This message should be given the lowest priority.

#### <sup>3</sup> ContentType Values

Value	Gateway Constant Name	Description
0	CONTENT_TYPE_TEXT	Text/plain - a plain text message
1	CONTENT_TYPE_HTML	Text/html - an HTML message
2	CONTENT_TYPE_XML	Text/XML - an XML message
3	CONTENT_TYPE_MULTIPART	multipart/alternative - a multipart message (contains both text and HTML)

#### <sup>4</sup> Status Values

Value	Gateway Constant Name	Description
0	STATUS_UNSENT	Message has not been sent yet.
1	STATUS_SENDING	Message is being sent now.
2	STATUS_SENT	Message sent successfully.
3	STATUS_PENDING	Message has not been sent yet and do not send it for now
4	STATUS_PICKEDUP	Message has been picked up by a Message Sender, Message Sender has not attempted to send yet
99	STATUS_ERROR	Error sending message, see ErrorText column for details.

### 23.14 MessageQueueFiles

This table stores binary or textual data associated with messages in the MessageQueue table. The binary or textual data may be handled differently depending on the kind of message in the MessageQueue table. Currently only email messages are stored in that table, so the Message Sender Service treats data stored in this table as Email Attachments.

#### Columns

Column	Type	Allow Nulls	Description
MessageQueueFileID	Integer	N	Primary key, always unique
MessageQueueID	Integer	N	Foreign key to MessageQueue.MessageQueueID
FileName	NVarchar(128)	N	Name associated with this data
FileSize	Integer	N	Size of data in FileData column
FileData	Image	Y	Binary or textual data
FileType	Integer	Y	Indicates how the file should be processed by Message Sender. <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKMsgQFMessageQueueFileID	P	MessageQueueFileID	Primary key.
IXMsgQFMessageQueueID		MessageQueueID	Used by the query to get files for a message queue entry

<sup>1</sup> FileType Values

Value	Gateway Constant Name	Description
0	mqftAttachment	The file is an attachment
1	mqftEmail	The file is the contents of the email. The email contents will be loaded from this file instead of the fields from the MessageQueue table.

### 23.15 Notes

This table contains general memo text for several Galaxy modules.

#### Columns

Column	Type	Allow Nulls	Description
NoteID	Int	N	Primary key, always unique.
NoteType	Char(10)	Y	Indicates how, and by whom, the Note is used. <sup>1</sup>
OwnerID	Int	Y	Foreign key to the table indicated by the NoteType. <sup>1</sup> May be null if only one note per owner is permitted.
NoteLineNo	Int	Y	The line number of this record. Always 1. When in French Fiscal Mode this is the Node Number.
NoteText	Text	Y	The text of the note.
NonUrgent	Bit	Y	Determines if the note is urgent
NoteGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKNotesNoteID	P	NoteID	Primary Key.
IXNotesOwnerID		OwnerID	Used to select the notes for a particular "owner".

<sup>1</sup> NoteType Values

Value	Description	OwnerID
CUSTNOT	Customer Note.	Foreign key to Customers.CustomerID.
EVTNOT	Event Note.	Foreign key to Events.EventID.
INVCNOT	Invoice Note.	Foreign key to ARInvoiceLines.InvoiceID.
ORDNOT	Order Note.	Foreign key to Orders.OrderID.
LINENOTE	Order Line Note.	Foreign key to OrderLines.OrderLineID.
NODESTAT	Node use state change note.	

## 23.16 REConfig

Configuration options required for the Raiser's Edge interface are stored in this table.

### Columns

Column	Type	Allow Nulls	Description
REConfigID	Int	N	Primary key, always unique
Name	Varchar(128)	N	Name of the Raiser's Edge config record. User can put the name of the Raiser's Edge database here. The value in this column is for display/identification purpose only
LicenseKey	Varchar(30)	N	License key for the Raiser's Edge interface
SerialNo	Varchar(30)	N	Serial number for the Raiser's Edge interface
Username	Varchar(20)	N	Username, used to connect to the Raiser's Edge database
Password	Varchar(20)	N	Password, used to connect to the Raiser's Edge database
REDatabase	Int	N	The Raiser's Edge database (value is actually the database ID)
UseConstituents	Bit	N	Set when constituents needs to be used
SendUsage	Bit	N	Set when Galaxy turnstile controller needs to send usages to the Raiser's Edge
UseVisualIDs	Bit	N	Set when Galaxy needs to generate VisualIDs. If not set then the Raiser's Edge generated VisualIDs are used
Server	Bit	N	Set when the Raiser's Edge interface is used in server mode. For example, POS and eGalaxy both use RE interface in server mode so this option is always set to 1
BarcodeLength	Int	Y	Length of the barcode
CIDStart	Int	Y	CID start value
CIDLength	Int	Y	CID length
MIDStart	Int	Y	MID start value
MIDLength	Int	Y	MID length
ProgramStart	Int	Y	Program start value
ProgramLength	Int	Y	Program length
AccessStart	Int	Y	Access start value
AccessLength	Int	Y	Access length
CheckDigitStart	Int	Y	Check digit start value
CheckDigitLength	Int	Y	Check digit length
MemberTableStart	Int	Y	MemberTable start value
MemberTableLength	Int	Y	MemberTable length
CardNumberStart	Int	Y	CardNumber start value
CardNumberLength	Int	Y	CardNumber length
VisualIDType	Int	Y	This field determines how the visual ID will be constructed for a membership in RE. A value of 0 means the constituent ID will be used. A value of 1 means the visual ID will be constructed from the barcode fields.
AdditionalMembersOption	Int	Y	This determines how additional members will be handled by RE Import and RE Merge. <sup>1</sup>
PassMembershipField	Int	Y	This field determines which pass user field will be used by RE Merge to store the Raiser's Edge membership ID on the Galaxy Pass. A value of 0 means no field will be used and values of 1 to 10 indicate the pass user field that will store the membership ID.
PassConstituentField	Int	Y	Holds Raiser's Edge Constituent ID
JointMembersAsConstituents	Bit	Y	Determines if the "Is constituent" flag is turned on for the Raiser's Edge record when merging a joint member contact.
HomePhoneTypes	nvarchar(250)	Y	Comma-separated list of phone types from the Raiser's Edge that will be used to map a constituent phone record to the Phone field on the corresponding CustContact in Galaxy.
EmailPhoneTypes	nvarchar(250)	Y	Comma-separated list of phone types from the Raiser's Edge that will be used to map a constituent phone record to the Email field on the corresponding CustContact in Galaxy.
CellPhoneTypes	nvarchar(250)	Y	Comma-separated list of phone types from the Raiser's Edge that will be used to map a constituent phone record to the Cell field on the corresponding CustContact in Galaxy.
FaxPhoneTypes	nvarchar(250)	Y	Comma-separated list of phone types from the Raiser's Edge that will be used to map a constituent phone record to the Fax field on the corresponding CustContact in Galaxy.
PassSubCategoryField	Int	Y	The number of the pass user field that will hold the Raiser's Edge membership subcategory description.
DefaultSubCategory	nvarchar(60)	Y	Contains the description default subcategory to add to the Raiser's Edge membership when the Galaxy membership has a blank subcategory value.
KeepSameREMembership	Bit	Y	A value of 1 means that when renewing, reissuing, or upgrading a pass in Galaxy where a new pass record is generated, no new membership will be generated in the Raiser's Edge. The new pass record will be linked to the existing membership in the Raiser's Edge.
MarkGiftsDoNotPost	Bit	Y	Set to 1 to mark all gifts inserted into the Raiser's Edge with a GL post status of "Do Not Post"
UseSubCatToMapPassKind	Bit	Y	A value of 1 indicates that when merging a membership from the Raiser's Edge to Galaxy, the data in the subcategory field must match the subcategoryID on the pass kind. The pass kind will only be considered a match if the program/category and subcategory all match exactly. When merging a pass from Galaxy to the Raiser's Edge, this setting will cause the subcategory on the pass kind to be used instead of the subcategory from the pass user field (if one is configured).
FundRestrictionMode	Integer	Y	Indicates the fund restriction mode for determining which gifts are merged and which are restricted.
FundDescriptions	nvarchar(max)	Y	Comma-separated list of fund descriptions used to determine which funds are included or excluded from gift merging based on the fund restriction mode. The fund descriptions are case-insensitive, and they support wildcard characters (*) for matching.
PreventChainMerge	Bit	Y	A value of 1 indicates that REConsole will not merge the relationships for a constituent while merging the constituent. Instead, GxTrigger records will be added for the relationships, so they can be added later. This is an option that must be set directly in SQL, as there is no setting for it in REConsole.

### Indexes

Name	Kind	Columns	Purpose
PKREConfigID	P	REConfigID	Primary Key.

<sup>1</sup> AdditionalMembersOption Values

Value	Gateway Constant Name	Description
0	AMO_NONE	No support for additional members
1	AMO_USER_FIELDS	Additional members will be added through pass user fields in Galaxy and will be added in RE as membership cards. The results of this setting are the equivalent of what happened in the old Galaxy RE interface when the "Additional members as constituents" setting was not checked (UseConstituents = False).

### 23.17 RFIDMaps

The table creates a link between the RFID tags sold at the POS and a visualid. The *ActiveCard* field denotes which VisualID is the current VisualID for the RFID tag since the tags can be re-used.

#### Columns

Column	Type	Allow Nulls	Description
RFIDMapID	Int	N	Primary key, always unique
VisualID	Varchar(40)	N	Visual ID
SerialNo	Char (20)	N	Serial number of the RFID tag
ActiveCard	Bit	N	Is this the current Visual ID
RFIDType	Int	Y	Indicates the type of RFID that this is a mapping for. Ex. Debit, Ticket <sup>1</sup>

#### Indexes

Name	Kind	Columns	Purpose
PKRFIDMapID	P	RFIDMapID	Primary Key.
IXRFIDMapsSerialNo	IX	SerialNo	Speed up querying.
IXRFIDMapsType	IX	RFIDType	Allows loading of RFIDMaps records based on type.
IXRFIDMapsActiveCard	IX	ActiveCard	Speed up querying.

<sup>1</sup> RFIDType Values

Value	Gateway Constant Name	Description
0	RFID_MAP_TYPE_DEBIT	Debit mapping
1	RFID_MAP_TYPE_TICKET	Ticket mapping

### 23.18 SecurityAnswers

This table contains the security answers provided by the user.

#### Columns

Column	Type	Allow Nulls	Description
SecurityAnswerID	Int	N	Primary key, always unique
UserID	Int	N	Reference to GxUsers.UserID
CodeTableValueID	Int	N	Foreign key to CodeTableValues
SecurityAnswerText	VarChar(255)	N	This is the security answer provided by the user
SecurityAnswerGUID	uniqueidentifier	N	Globally unique identification value - uniquely identifies a record across systems

#### Indexes

Name	Kind	Columns	Purpose
PKSecurityAnswerID	P	SecurityAnswerID	Primary Key.
IXSecurityAnswerUserID	IX	UserID	Index on user id

### 23.19 SecurityLog

The purpose of this table is to record when user view secure information. Currently it will be storing when users view encrypted credit card numbers.

#### Columns

Column	Type	Allow Nulls	Description
SecurityLogID	Int	N	Primary key, always unique
JnlTranID	Int	N	Link to the journal details table-the journal transaction that indicated a user viewed a secured data
UserID	Int	N	ID of the user who viewed the data
ViewDate	DateTime	N	The date when the user viewed the data
Node	Int	N	The node on which the date was viewed
ApplicationID <sup>1</sup>	Int	N	From which Application the date was viewed
NodeNumberViewed	Int	N	The node number of the transaction viewed
DateOfTransaction	DateTime	N	The date of the transaction viewed. This value defaults to 1 (1899-12-31 00:00:00.000) in the application, which means that the date of the transaction was unknown at the time of viewing.
TransactionNumber	Int	N	The transaction number of the transaction viewed
InformationViewed	Text	N	The data that was viewed (credit card numbers are masked)

#### Indexes

Name	Kind	Columns	Purpose
PKSecurityLogID	P	SecurityLogID	Primary Key.
IXSecurityLogUserID	IX	UserID	Index to access data by user
IXSecurityLogTransactionDateNumber	IX	DateOfTransaction, TransactionNumber	Index to access data by transaction viewed

#### <sup>1</sup> Application ID Values

Value	Gateway Constant Name	Description
0	LEGACY_APPLICATION_ID	Legacy - this value should not be in the table
1	GALAXY_APPLICATION_ID	Galaxy
2	MWS_APPLICATION_ID	MWS

## 23.20 StateRegions

This table contains a list of State, Region, or Province level sub-division per country.

Column	Data Type	Description
StateRegionID	Int	Primary key for this table.
Name	Varchar(128)	State or Region Name. i.e. Pennsylvania, New York, etc.
CountryCode	Char(2)	This is a ISO 3166 two letter Country Code. Foreign key to Countries.CountryCode
InActive	Bit	When this option is set to True (1), eGalaxy Web Store does not include this State for user selection.
DefaultLocale	Bit	Mark the State as the default state for a country. There can be only one Default Locale state per country. This is used in Web Store to pre-select the State for a country in a drop down list box.
Abbreviation	Varchar(50)	Abbreviation used for State / Region. i.e. PA for Pennsylvania, United States.
StateRegionGUID	uniqueidentifier	N

### Indexes

Name	Kind	Columns	Purpose
PKStateRegionID	PK	StateRegionID	Table primary key
IXStateRegionsCountryCodeName	IX	CountryCode, Name	Unique Index to prevent duplicated State name to be entered for a country.

### 23.21 TicketLog

Column	Type	Allow Nulls	Description
TicketLogID	Int	N	
LogKind	Int	N	
EntryDateTime	DateTime	N	
AgencyID	Int	N	
NodeID	Int	N	
AgentID	Int	N	
TransID	Int	N	
VisualID	Char(40)	N	
Serial	Char(20)	N	
ReasonID	Int	N	
LockoutAdded	Bit	Y	
TicketPresent	Bit	Y	
TicketAmount	Money	Y	

### Indexes

Name	Kind	Columns	Purpose
PKTicketLogTicketLogID	PK	TicketLogID	Primary Key always unique
IXTicketLogCancelTktReasonRep	IX	ReasonID, VisualID, AgencyID, EntryDateTime, AgentID, NodeID	Improve performance of Ticket Cancel Report

## 24 Auto Upgrade Database

This section lists the tables required by the GalaxyDatabase.SQL script.

The GalaxyDatabase script automatically detects any table changes on current Galaxy database version to the upgrade version and upgrades the database.

## 24.1 ManualUpgradeTables

This table contains the tables that are to be skipped during the GalaxyDatabase script run. This allows the large tables to be upgrade separately. On each run of the GalaxyDatabase script, the Manual column gets set to '1' at the end of the script run.

### Columns

Column	Type	Allow Nulls	Description
ManualUpgradeTableID	Integer	N	Always unique, primary key
TableName	Varchar(128)	N	Name of the table for the GalaxyDatabase to skip the upgrade
Manual	Bit	N	When set to 1, this table will be skipped during the upgrade

### Indexes

Name	Kind	Columns	Purpose
PKManUpgradeTblIDManUpgrdTblID	P	ManualUpgradeTableID	Primary key.
AXManualUpgradeTablesTableName	A	TableName	Unique key to make sure there is only one table name entry

## 24.2 TableIndexDefs

This table contains the index definitions. This table is used to recreate the dropped indexes during the upgrade script run.

### Columns

Column	Type	Allow Nulls	Description
TableName	Varchar(128)	N	Name of the table the index belongs to
IndexName	Varchar(128)	N	Name of the index
IndexColumns	Varchar(256)	N	The columns that defines the index. For a multiple-columned index, the each column is separated by a comma, without any blank spaces in between.
IsPrimaryKey	Bit	N	When set to 1, this index is a primary key
IsConstraint	Bit	N	When set to 1, this index is a constraint
IsClustered	Bit	N	When set to 1, this index is a clustered index
IsUnique	Bit	N	When set to 1, this index is an unique index
Dropped	Bit	N	When set to 1, this index has been dropped by the upgrade script
Recreated	Bit	N	When set to 1, this index has been recreated by the upgrade script

### Indexes

Name	Kind	Columns	Purpose
IXTableIndexDefsTableName		TableName	To speed up lookups when recreating index
IXTableIndexDefsIndexName		IndexName	To speed up lookups when recreating index

### 24.3 UpgradeDefaultValues

This table contains the default values to use when a column needs to be changed from NULL to NOT NULL.

#### Columns

Column	Type	Allow Nulls	Description
TableName	Varchar(50)	Y	Name of the table. When NULL, every table is selected when selected for a default value
ColumnName	Varchar(50)	Y	Name of the column. When NULL, every column within the TableName is selected for a default value
ColumnType	Char(20)	Y	Data type of the column. When NULL, every column type for the ColumnName within the TableName is selected for a default value
DefaultValue	Varchar(100)	N	Default value to use for TableName, ColumnName, and ColumnType match

## 24.4 UpgradeHistory

This table contains the history of upgrade of database via GalaxyDatabase.SQL script. Each record has the Galaxy version of the upgrade script, upgrade settings, and whether the script run was successfully or not.

The last entry in this table is the current database version.

### Columns

Column	Type	Allow Nulls	Description
UpgradeHistoryID	Integer	N	An Identity column to make a row unique.
GalaxyVersion	Char(10)	N	Galaxy version of the GalaxyDatabase.SQL script
RunMode	Int	N	Run mode of the GalaxyDatabase script <sup>1</sup>
RunFromHost	Varchar(128)	N	The name of the host machine connected to run the upgrade script
RunDate	DateTime	N	Date of the upgrade script run
ErrorID	Int	N	Error encountered during the run <sup>2</sup>
Log	Varchar(4000)	N	Contains the Upgrade setting of the upgrade script run. Also, if an error occurred during the upgrade script run, the error message is appended to the end.

### <sup>1</sup> RunMode Values

Value	Description
11	Upgrade script was ran in Upgrade mode
21	Upgrade script was ran in Post Upgrade mode

### <sup>2</sup> ErrorID Values

Value	Description
< 0	The error was detected by the upgrade script
> 0	The error was detected by the MS SQL server
0	No error encountered
-1	Upgrade found a dependency while trying to upgrade a column but upgrade setting was not set to automatically drop the dependencies. The upgrade script has been terminated
-2	Trying to modify a column but the column does not exists in the table
-3	Upgrade script needs to change the column from NULL to NOT NULL but it could not find a default value for this column in UpgradeDefaultValues table. This ErrorID should not occur

## 24.5 UpgradeLogs

This table contains the upgrade scripts produced by the GalaxyDatabase.SQL script. This table gets truncation on each run of the GalaxyDatabase script if it is configured to do so.

### Columns

Column	Type	Allow Nulls	Description
Log	Varchar(4000)	Y	The upgrade script or the description of the upgrade script
LogDate	DateTime	Y	Date and time then the Log was inserted

## 24.6 UpgradeSettings

This table is used during the database upgrade using the GalaxyDatabase.SQL script. This table stores the current run of the upgrade settings to control the upgrading progress.

### Columns

Column	Type	Allow Nulls	Description
Setting	Varchar(50)	N	Type of header record, see table. <sup>1</sup>
Value	Varchar(128)	N	Setting value

### <sup>1</sup> Setting Values

Value	Description
RunMode	<p>Running mode of the GalaxyDatabase script.</p> <p>The value of the Value column</p> <p>'10': (default) Run as Upgrade Test mode, upgrade scripts are NOT executed</p> <p>'11': Run as Upgrade mode, upgrade scripts ARE executed</p> <p>'20': Run as Post Upgrade Test mode, post upgrade scripts are NOT executed</p> <p>'21': Run as Post Upgrade mode, post upgrade scripts ARE executed</p>
Database	<p>The Value column indicates the name of the database being upgrade. If the name of the database is not the same as the connected database, the upgrade script will terminate with an error.</p>
DropDependency	<p>This setting allows or disallows the automatic dropping of all the dependent indexes when a column needs to be upgraded.</p> <p>The value of the Value column</p> <p>'0': (default) Do not automatically drop the dependent indexes. The upgrade script will terminate with an error when it encounters a column dependency.</p> <p>'1': Do drop the dependent indexes automatically.</p>
RecreateDroppedIndexes	<p>This allows or disallows the script to automatically recreate the dropped indexes via DropDependency upgrade setting above.</p> <p>The value of the Value column</p> <p>'0': (default) Do not recreate the dropped indexes. The script to recreate the dropped columns will be supplied but it must be manually executed.</p> <p>'1': Do recreate the dropped indexes. Currently, recreate of index does not support FILLER option.</p>
DropObsoleteColumns	<p>This allows or disallows the script to drop the obsolete columns.</p> <p>The value of the Value column</p> <p>'0': (default) Do not drop the obsolete columns. The script will continue even if it encounters obsolete columns. The script to drop the column will be provided but will not be executed.</p> <p>'1': Do drop the obsolete columns.</p>
TruncateUpgradeLogs	<p>The GalaxyDatabase script logs all the upgrade script to the UpgradeLog table. This setting allows or disallows the upgrade script to truncate the table on each run, so that the UpgradeLog contains the current upgrade scripts.</p> <p>The value of the Value column</p> <p>'0': (default) Keep all the upgrade scripts.</p> <p>'1': Truncate the UpgradeLog table on each run</p>

**25 External Application**

## 25.1 BatchTicketImports

The **BatchTicketImports** table records the tickets and/or passes that were added, changed, or returned as part of a batch process using the TKTIMPRT program.

### Columns

Column	Type	Allow Nulls	Description
BatchTicketImportID	Int	N	Primary key, always unique.
ProcessID	Int	N	Process identifier from the Header record of the batch file
ProcessDate	DateTime	N	The date and time the batch file was created
VisualID	Char(40)	N	The Visual ID number of the ticket or pass
AccessCode	Int	Y	The Access Code for the ticket or Pass (may be zero if an error prevented the Access Code from being looked-up).
TableKind	Int	N	Indicates the table to which this record refers <sup>1</sup>
Status	Int	Y	This indicates whether or not the ticket or pass was imported, or rolled-back when the batch was last processed. <sup>2</sup>
ImportResult	Int	Y	A number representing the result of the import process. <sup>3</sup>
RollbackResult	Int	Y	A number representing the result of the rollback process. <sup>3</sup>

### Indexes

Name	Kind	Columns	Purpose
PKBatchTicketImportsBTktImpID	P	BatchTicketImportID	Primary key.
IXBatchTktImpACVIDPIDPDateIRRR		AccessCode, VisualID, ProcessID, ProcessDate, ImportResult, RollbackResult	Ticket Import application produces a report after importing each file, this index would be used while retrieving data for that report.
IXBatchTktImpPIDPDate		ProcessID, ProcessDate	While processing 'Returns' in the Ticket Import application a query is made to the table, this index would be used by that query and hence the performance of application would improve.

#### <sup>1</sup> TableKind Values

Value	Constant Name	Description
0	TBLKIND_TICKETS	The Tickets table is used with this record
1	TBLKIND_PASSES	The Passes table is used with this record

#### <sup>2</sup> Status Values

Value	Constant Name	Description
0	STATUS_IMPORT	The ticket/pass was imported on the last run of this batch
1	STATUS_ROLLBACK	The ticket/pass was rolled-back on the last run of this batch

#### <sup>3</sup> Import Result and Rollback Result Code Values

Value	Constant Name	Description
0	RESULT_SUCCESS	Processing was successful
1	RESULT_NO_ACCESS_CODE	A record for the ticket's Access Code is not in the access code table
2	RESULT_BAD_VISUAL_ID	The Visual ID number is blank, or cannot be matched to a media definition
3	RESULT_BAD_STATUS	The status code is not valid
4	RESULT_TICKET_HAS_USAGE	The ticket/pass has usage and cannot be returned
5	RESULT_NO_TICKET	The Visual ID has a Media Def and Access Code but no ticket
89	RESULT_ERROR	A miscellaneous error code (e.g., memory allocation error)
98	RESULT_MULTI_VISUAL_ID	This is a success-with-warning. The processing was successful, but there are multiple tickets/passes for the specified Visual ID. Visual ID's are unique, and should never be duplicated, so there may be a problem with the table.
99	RESULT_NOT_RUN_YET	A rollback process has not been performed (rollback result only)

- Note: Result codes above 90 are warning messages. Codes between 1 and 89 inclusive are error messages.

## 25.2 COAReportExport

This table contains the exported data from Chart Of Account based reporting. Currently, Composite Income and Deferred Revenue reports are exported to these tables.

### Columns

Column	Type	Allow Nulls	Description
COAReportExportID	Int	N	Primary key, always unique
ReportID	Int	N	Report the date was extracted from. <sup>1</sup>
WorkingDate	DateTime	N	Working Date of sale or usage
AccountID	Char(12)	N	Chart Of Account. This is a concatenation of  Company Number zero padded to 4 characters  General Ledger code zero padded to 3 characters (i.e. 101 for ticket)  Category zero padded to 3 characters  Sub-Category zero padded to 2 characters
ExternalAccount	VarChar(30)	N	User defined account number outside of Gateway Applications
Quantity	Int	N	Number of journal or usage records included in this one COAReportExport record
Amount	Money	N	Total Amount for Quantity
Posted	DateTime	Y	Date this record was posted by the external application. Galaxy application does not set this value. The external application must set this value to indicate that the record has been processed by it.  Galaxy will not set this value. Hence, this value will be NULL when Galaxy creates this record.

### Indexes

Name	Kind	Columns	Purpose
PKCOAReportExportID	P	COAReportExportID	Primary Key.
IXCOAReportExportReportIDPostd		ReportID, Posted	Index to speed up the filter performance by Report ID
IXCOAReportExportReplDWDAcctID		ReportID, WorkingDate, AccountID	Index used during update

<sup>1</sup> ReportID Values

Value	Gateway Constant Name	Description
101	AGENCY_COMPOSITE_INCOME_REPORT_ID	Agency Composite Income Report
121	AGENCY_DEFERRED_REVENUE_REPORT_ID	Deferred Revenue Report

### 25.3 OiRejectedFileNames

This table contains the reasons for the input file rejection by OrderImports application due to an invalid file name. The invalid file name includes a wrong file name length (must be 21 characters long without the extension), and the first three digits of the input file name must be numeric and it must be in a range of 1-600, inclusive.

#### Columns

Column	Type	Allow Nulls	Description
OiRejectedFileNameID	Int	N	Primary key, always unique.
FileName	Char(30)	N	The file name of the input file which being rejected.
LotDate	DateTime	N	The LotDate of the input file. This value is from the header record of the input file.
FileCreateDate	DateTime	N	The date the input file was created.
RejectDate	DateTime	N	The date of the input file reject.
RejectReason	Char(50)	N	The reason for the rejection.
HeaderLine	Char(120)	Y	The first line of the input file

#### Indexes

Name	Kind	Columns	Purpose
	P	OiRejectedFileNameID	Primary Key.
OiRejectedFileNames_LotDate		LotDate	For selecting the report by the LotDate
OiRejectedFileNames_RejectDate		RejectDate	For selecting the report by the RejectDate

## 25.4 RejectedOrderHeader

This table stores the header information for the RejectedOrders.

### Indexes and Constraints

Primary Key: PKRejectedOrderHeaderID

Indexes:

(None)

Column	Type	Allow Nulls	Description
RejectedOrderHeaderID	Int	N	Primary key, always unique
LotID	Int	N	LotID for this reservation
LotDate	DateTime	N	LotDate for this reservation
CheckinDate	DateTime	N	Check-in date for this reservation
CustomerCategory	Char (3)	N	Customer Category or Delivery Code for this reservation
Customer	Char (45)	N	Name of the customer for this reservation
Origin	Char (20)	N	Distribution channel for this reservation
NodeNo	Int	N	Node for this reservation
UserName	Char (20)	N	Name of the user

## 25.5 RejectedOrderDetails

This table stores the detail information for the RejectedOrders.

### Indexes and Constraints

Primary Key: PKRejectedOrderDetailID

Indexes:

(None)

Column	Type	Allow Nulls	Description
RejectedOrderDetailID	Int	N	Primary key
RejectedOrderHeaderID	Int	N	Foreign key that references to the RejectedOrderHeader table
OrderID	Char(20)	N	The ReservationID
RejectionReason	Char (80)	N	Reason for the rejection of this reservation
RoomSequenceID	Int	N	Room sequence number for this reservation
RegisterNo	Char(5)	N	Register number for this reservation
GuestID	Char (3)	N	PAX id or Guest id for this reservation
GuestName	Char (30)	N	PAX Name of Guest name for this reservation
LeadGuestName	Char (30)	N	Lead PAX Name or Lead Guest Name for this reservation
PLU	Char (20)	N	Ticket code or PLU for this reservation
TicketStartDate	DateTime	N	Start Validity date for ticket for this reservation
TicketEndDate	DateTime	N	End Validity date for ticket for this reservation
Qty	Int	N	No. Of tickets for this reservation
Price	Currency	N	Price of ticket
CancellationFlag	Int	N	The Status <sup>1</sup> for this reservation
PrintSequenceNo	Int	N	Printing sequence number
Corrected	Bit	N	Flag <sup>2</sup>
Filler1	Char (20)	Y	Not currently used
Filler2	Char (10)	Y	Not currently used
Filler3	Char (10)	Y	Not currently used

<sup>1</sup> Status Values

Value	Description
01	Reservation Issue
02	Reservation Update
03	Reservation Cancel

<sup>2</sup> Corrected Flag Values

Value	Description
TRUE	Reservation has been received correctly later
FALSE	By default False and reservation has not been received correctly

## 25.6 UserDiscrepancyLog

This table stores the UserID discrepancy between an external application and Galaxy.

The table is created for the DLP (Dev025 - Allegro Interface). A row will be added to this table whenever the user logged on to any external system and the user logged on to Galaxy is different. In DLP's case the external application is Allegro System.

### Indexes and Constraints

Primary Key: PKUSERDISCREPANCYLOGID

Indexes:

(None)

Column	Type	Allow Nulls	Description
UserDiscrepancyLogID	Int	N	Primary key, Always unique
LogDate	DateTime	N	Date and time when the User Discrepancy happened
AgencyID	Int	N	Agency where the User Discrepancy happened
NodeID	Int	N	Node where the User Discrepancy happened
GalaxyUserID	Int	N	User logged on to Galaxy
ExternalUserID	Int	N	User logged on to the external application (e.g Allegro)

## 26 Utilities

This section outlines miscellaneous utilities within the GalaxyDatabase.SQL script.

## 26.1 GTS\_SP\_CopyDatabase

Creates a duplicate of the Galaxy Database on the same server that can be used for testing, reporting, or simple backup.

### Parameters

Column	Type	Allow Nulls	Description
@TempBackupPath	nvarchar(255)	N	A path on the local disk to be used to store a temporary file while the procedure is running.
@CopyDatabaseName	Sysname	N	The desired name of the database copy to be created.
@DataFilePath	nvarchar(255)	N	The path on the local disk where the new data file should be placed.
@LogFilePath	nvarchar(255)	N	The path on the local disk where the new log file should be placed.
@Force	Bit	N	If set to 1, and the copy database already exists, the procedure will disconnect all existing connections before performing the restore.
@PostRestoreCmd	nvarchar(max)	Y	An optional stored procedure to be executed after the restore process is complete. This must be a stored procedure with no required parameters. This is useful for granting more rights or creating custom indices on the reporting database. This should be specified as "<schema>.<procedure>".
@Help	Bit	N	Shows the stored procedure help information

### Examples

#### Call:

```
USE GalaxyProd
EXEC GTS_SP_CopyDatabase
'C:\Program Files\Microsoft SQL Server\MSSQL10_50.SQLEXPRESS\MSSQL\Backup'
, 'GalaxyTraining'
, 'C:\Program Files\Microsoft SQL Server\MSSQL10_50.SQLEXPRESS\MSSQL\Data'
, 'C:\Program Files\Microsoft SQL Server\MSSQL10_50.SQLEXPRESS\MSSQL\Data'
, 1
, null
, 0
GO
```

#### Result:

Copies the GalaxyProd database and creates a database named GalaxyTraining. Galaxy training uses the following files:

- C:\Program Files\Microsoft SQL Server\MSSQL10\_50.SQLEXPRESS\MSSQL\Data\GalaxyTraining.mdf
- C:\Program Files\Microsoft SQL Server\MSSQL10\_50.SQLEXPRESS\MSSQL\Data\GalaxyTraining.ldf

If GalaxyTraining already exists and has connections, those connections will be dropped during the copy process.

During the execution of the procedure, a temporary file is placed in C:\Program Files\Microsoft SQL Server\MSSQL10\_50.SQLEXPRESS\MSSQL\Backup.

## 26.2 GTS\_SP\_PurgeGXRecordLog

This procedure removes entries from the GXRecordLog table that are beyond a specified age. By automating the use of this procedure, you can prevent Galaxy's Audit Logging capability from becoming an unnecessarily large part of your database.

### Parameters

Column	Type	Allow Nulls	Description
@RetentionPeriodInDays	Smallint	N	Sets the maximum age of a record that should remain after the purge.
@Help	Bit	N	Shows the stored procedure help information

### Examples

Call:

```
USE GalaxyProd
EXEC GTS_SP_PurgeGXRecordLog
90
, 0
GO
```

### Result:

Deletes audit log records from the GXRecordLog table that are over 90 days old from the database.

## 26.3 GTS\_SP\_PurgeRowChanges

In a centrally managed environment, the RowChanges table tracks all modifications made to centrally managed data and makes those changes available for nodes when published. In an active system this table can become quite large. This stored procedure provides a way to remove old entries that are no longer needed, and can be automated through a variety of tools such as SQL Agent.

### Parameters

Column	Type	Allow Nulls	Description
@DateOrPublishCount	Varchar(15)	N	Indicates if the data should be purged based on the date of the change or the distance from the most recent publish. <sup>1</sup>
@RetentionPeriodInDays	Smallint	N	If using Date mode, this is the maximum age of a record should remain after a purge.
@PublishesToKeep	Smallint	N	If using PublishCount mode, if a record is associated with a publish number that is further from the current publish by more than this value, it will be purged. For example, if PublishesToKeep is set to 2, and the current publish number is 20, any records for publishes before publish 18 will be purged.
@SetOutOfDateNodesForResynch	Bit	N	With this option set to 1, if a node has not downloaded and applied changes for rows that are being purged, it will be set to automatically perform a refresh at the next connection.  For example, if purging by publish count, where the current publish number is 20 and PublishesToKeep is set to 2, a node that is currently on publish 15 will be set to refresh.  This ensures that data changes that are purged are not missed by the nodes when they reconnect to the database.
@Help	Bit	N	Shows the stored procedure help information

<sup>1</sup> DateOrPublishCount Values

Value	Effect
'Date'	Sets the stored procedure to purge based on the maximum age of a record to be retained. Use this in conjunction with the @RetentionPeriodInDays parameter.
'PublishCount'	Sets the stored procedure to purge based on the maximum number of publishes to be retained. Use this in conjunction with the @PublishesToKeep parameter.

### Examples

Call:

```
USE GalaxyProd
EXEC GTS_SP_PurgeRowChanges
'PublishCount'
, 0
, 5
, 1
, 0
GO
```

#### Result:

Purges any records in the RowChanges table associated with publishes more than 5 publishes behind the current publish number. If a node's current publish is more than 5 publishes behind, it will be set to refresh all data.

Call:

```
USE GalaxyProd
EXEC GTS_SP_PurgeRowChanges
'Date'
, 90
, 0
, 1
, 0
GO
```

#### Result:

Purges any records in the RowChanges table that are more than 90 days old. If a node has not connected to the database in more than 90 days, it will be set to refresh all data.