



Salesforce.com: Winter '12

Database.com Winter '12 Release Notes



Last updated: October 15, 2011

© Copyright 2000–2011 salesforce.com, inc. All rights reserved. Salesforce.com is a registered trademark of salesforce.com, inc., as are other names and marks. Other marks appearing herein may be trademarks of their respective owners.

Table of Contents

Introduction.....	3
About the Release Notes.....	4
Summary of Winter '12 Features and Impact on Database.com Users.....	5
General Availability During Major Release Upgrades.....	8
Browser Support Enhancements.....	9
Database.com Console Enhancements.....	10
Schema Builder—Beta.....	10
Force.com Developer Console—Generally Available.....	11
Recycle Bin Now Stores Records for 15 Days.....	14
Unique Subdomain for the Database.com Console.....	14
Permission Sets—Generally Available.....	15
Overview of User Permissions and Access.....	15
Creating Permission Sets.....	16
Using Permission Set Lists.....	17
Working in a Permission Set's Overview Page.....	19
About App and System Permissions.....	20
Assigning Permission Sets.....	21
Searching Permission Sets.....	21
Using a Permission Set's Assigned Users Page.....	22
Revoking Permissions and Access.....	23
Security Enhancements.....	24
Apex Enhancements.....	25
API Enhancements.....	29
New and Changed Objects.....	30
Chatter API Objects.....	31
Metadata API Enhancements.....	33
REST API Enhancements.....	34
Streaming API—Pilot.....	35
Chatter REST API Enhancements.....	36
New and Changed SOAP Calls.....	41

Database.com Java SDK—Beta.....	42
Additional Database.com Enhancements.....	43
Help and Training Enhancements.....	44

Introduction

Welcome to Database.com!

Database.com is a multitenant cloud database service that's designed to store data for mobile, social, web, and desktop applications. You can use Database.com as the back-end database for applications that are written in any language, and run on any platform or mobile device. Database.com's built-in social computing infrastructure and native support for building sophisticated REST-based APIs enables you to create employee-facing, native mobile and social apps. With Database.com, you can:

- Quickly and easily create schemas using Database.com's metadata-driven data modeling tools
- Store small to very large data sets, scaling up to millions of records
- Secure your data and share it only with certain people, whether they're within your company, your customers, or your partners
- Deliver Master Data Management by leveraging integration connectors from third-party vendors such as Informatica, IBM, and Progress Software

Because Database.com is in the cloud, and managed by salesforce.com, you don't need to worry about buying costly hardware and software, or managing the labor-intensive and specialized workarounds, scaling, tuning, backups, and upgrades that are required with traditional client/server databases, whether hosted or on premises. Additionally, managing user access to data is simplified because Database.com leverages Salesforce's proven identity and authentication model, as well as its sharing and security engine.

Database.com Features

- A metadata-driven data model for both structured and unstructured data
- Salesforce Object Query Language (SOQL) and Salesforce Object Search Language (SOSL) for querying your data
- Internal database services, such as the ability to create formulas, validation rules, and workflow
- Open REST and SOAP APIs for accessing and manipulating data
- The Apex programming language for extending your database with triggers, stored procedure classes, and custom Web services
- A user identity and authentication model, leveraging OAuth and SAML, with tightly integrated controls for data security, sharing, and social applications
- Chatter functionality, including Chatter feeds and social data such as users, groups, followers, and files, which you can add to your application

About the Release Notes

The Release Notes are a comprehensive user guide for the latest release of Database.com. Unlike a traditional release notes document that includes only a simple list of enhancements, the Database.com Release Notes give you everything you need to get up and running with the new features and enhancements in the latest release.

What's Included in the Release Notes

For every new major enhancement, the Release Notes provide:

- A brief, high-level description of the functionality
- Implementation tips to help you get started with setup and administration
- Best practice tips to help you maximize the benefit of the functionality
- Complete end-to-end instructions on how to set up and use the functionality

Beyond the major new features, the Additional Enhancements sections include a list and brief description of every other enhancement or functional change included in the latest release.

Let the Release Notes be your guide to success with the latest release from salesforce.com!

Your Feedback Matters

We know how important the Release Notes, online help, and documentation are to your company's success with Database.com. To continually improve the content we deliver to you, we want to know what works and what doesn't. Let us know!

- Feedback forms—Every HTML documentation page, both in the online help and in our developer guides at [Developer Force](#), includes a feedback form for you to submit your suggestions, corrections, and feedback about the documentation. Let us know what you think!
- IdeaExchange—We're listening to your ideas too. Winter '12 includes some of your top ideas. Visit [IdeaExchange](#) for a complete list of ideas coming in Winter '12.

Want to be notified whenever we publish new documentation or make significant updates to existing documentation? Follow us on Twitter: [@salesforcedocs](#).

Summary of Winter '12 Features and Impact on Database.com Users

Winter '12 has features that immediately impact all users after the release. You might want to communicate these changes to your users beforehand so they are prepared. Other features require direct action by an administrator before users can benefit from the new functionality.

The following table summarizes the Winter '12 features and their impact on users.

General Enhancements

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available requires some setup.	Contact salesforce.com to enable this feature.
General Availability During Major Release Upgrades	✓			

Database.com Console Enhancements

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available requires some setup.	Contact salesforce.com to enable this feature.
Schema Builder—Beta	✓			
Force.com Developer Console—Generally Available			✓	
Recycle Bin Now Stores Records for 15 Days	✓			
Unique Subdomain for the Database.com Console	✓			

Security

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available but requires some setup.	Contact salesforce.com to enable this feature.
Permission Sets—Generally Available	✓			
Updated Encrypted Custom Field Permission		✓		

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available but requires some setup.	Contact salesforce.com to enable this feature.
Revoking Remote Access Tokens Using OAuth API			✓	
New Chatter API Endpoint URLs For Identity Service			✓	
		✓		
New scope Parameter in OAuth			✓	
Single Sign-On Changes		✓		

Apex

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available but requires some setup.	Contact salesforce.com to enable this feature.
Apex REST		✓		
API for Asynchronous Test Runs—Beta		✓		
Common Test Data Creation Methods in Public Test Classes		✓		
Higher Governor Limits in Some Categories		✓		
JSON Support		✓		
New Schema Methods		✓		
New System Methods: isBatch, isFuture, and isScheduled		✓		
ReadOnly Annotation—Generally Available		✓		
Additional Apex Enhancements		✓		

APIs and Toolkits

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available but requires some setup.	Contact salesforce.com to enable this feature.
API Enhancements	✓			
Database.com Java SDK—Beta		✓		

Additional Database.com Enhancements

Feature	Automatically visible to all users. No setup required.	Automatically visible to all administrators. No setup required.	Not automatically visible. Feature is available requires some setup.	Contact salesforce.com to enable this feature.
Database.com Light User License		✓		
Domain Name Flexibility		✓		

General Availability During Major Release Upgrades

Starting with Winter '12, Database.com will be generally available during major release upgrades.

General availability means you should only expect to experience up to a five minute disruption of service as your Database.com organization is upgraded. Users trying to access Database.com during this time receive an error message that the service is unavailable and they can log back in momentarily. In addition, users logged into Database.com during a major release upgrade may be temporarily logged out.

For more information and to see the system status of your Database.com organization, go to trust.database.com.

Browser Support Enhancements

Winter '12 includes enhanced browser support. The Database.com Console now supports the following browsers:

Browser	Comments
Windows® Internet Explorer® versions 6, 7, 8, and 9	Salesforce.com strongly recommends using Internet Explorer version 9 over versions 6, 7, and 8. Apply all Microsoft® hotfixes.
Mozilla® Firefox®, most recent stable version	Salesforce.com recommends using Firefox for best performance and makes every effort to test and support the most recent version. For configuration recommendations, see “Configuring Firefox” in the Salesforce online help.
Google® Chrome™, most recent stable version	Google Chrome applies updates automatically; Salesforce.com makes every effort to test and support the most recent version. There are no configuration recommendations for Chrome.
Apple® Safari® version 5.1.x	Supported on Windows XP and Mac OS X version 10.4 and later. There are no configuration recommendations for Safari.

Database.com Console Enhancements

See Also:

[Schema Builder—Beta](#)

[Force.com Developer Console—Generally Available](#)

[Recycle Bin Now Stores Records for 15 Days](#)

[Unique Subdomain for the Database.com Console](#)

Schema Builder—Beta



You asked for it! Schema builder is from an [idea](#) on the IdeaExchange.



Note: This release contains a beta version of schema builder that is production-quality but has known limitations. Schema builder is not supported on Windows® Internet Explorer® version 6 or earlier.

In Winter '12, you can now view a read-only diagram of the elements of your schema all in one window. Schema builder eliminates the need to click from page to page to find the details of objects, fields, and relationships in your schema. Each object includes details such as the field values, required fields, and how objects are related by displaying lookup and master-detail relationships. The drag-and-drop interface lets you easily move objects around and saves the layout of your schema any time you move an object.

See Also:

[Database.com Console Enhancements](#)

[Working with Schema Builder](#)

Working with Schema Builder













Note: This release contains a beta version of schema builder that is production-quality but has known limitations. Schema builder is not supported on Windows® Internet Explorer® version 6 or earlier.

To access schema builder, click **Schema Builder** in the Database.com console.

When working with schema builder:

- Click on an object and move it to any space on the canvas. Schema builder saves the layout of your schema any time you move an object.
- To automatically sort the layout of the objects in your schema, click **Auto-Layout**. Once you click **Auto-Layout**, you can't undo it.
- Depending on your preferences you can display either element names if you prefer system names or labels if you prefer text values.

- ◊ To display element names, click **View Options** > **Display Element Names**.
 - ◊ To display element labels, click **View Options** > **Display Element Labels**.
 - To show or hide relationships, click **View Options** > **Show/Hide Relationships**. You may want to hide relationships if your schema is taking too long to load.
 - Hover over relationship lines to show relationship details such as lookup and master-detail relationships.
 - Click  to quickly find a specific object in your list of objects.
 - Click the drop-down list to filter your list of objects in the sidebar:
 - ◊ All Objects
 - ◊ Selected Objects
 - ◊ Custom Objects
 - ◊ System Objects
 - To clear all selected objects in the sidebar, click **Clear All**. To select all objects, click **Select All**.
 - To collapse the sidebar, click . To expand it, click .
 - The legend:
 - ◊  indicates a lookup relationship
 - ◊  indicates a master-detail relationship
 - ◊  indicates a field is required on an object
 - To show or hide the legend, click **View Options** > **Show/Hide Legend**. To close the legend, click ✖ in the legend window.
 - To view the details of a field in a new window, click the element name or label.
 - To view object and field details in a new window, click  > **View Object in New Window**.
 - To zoom in, click . To zoom out, click .
-  **Note:** You can't save the level of zoom when closing schema builder.
- To close the schema builder and save the layout of your objects, click **Close**.

See Also:

[Schema Builder—Beta](#)

Force.com Developer Console—Generally Available

The Force.com Developer Console is composed of a set of related tools that allow you to look at your source code and review how it executes. It can also be used to monitor database events, workflow, callouts, validation logic, and other events that are saved in debug logs.

To access the Developer Console, open the System Log by clicking **Your Name** > **System Log**.

What Can You Use the Developer Console For?

You can use the Developer Console to investigate a number of aspects of your organization, including:

General Debugging and Troubleshooting

At the highest level, the Developer Console provides a mechanism to inspect executed requests. It provides a fine-grained log, allowing you to review every statement executed within a request, and a set of interactive panels that allow you to simulate an execution "step-through".

Source Code Editing

The Developer Console allows you to edit and create triggers and classes within the Developer Console itself, letting you inspect, trace, and edit your code all in the same tool. You can open any number of code objects and switch between them with a single click.

Performance Validation

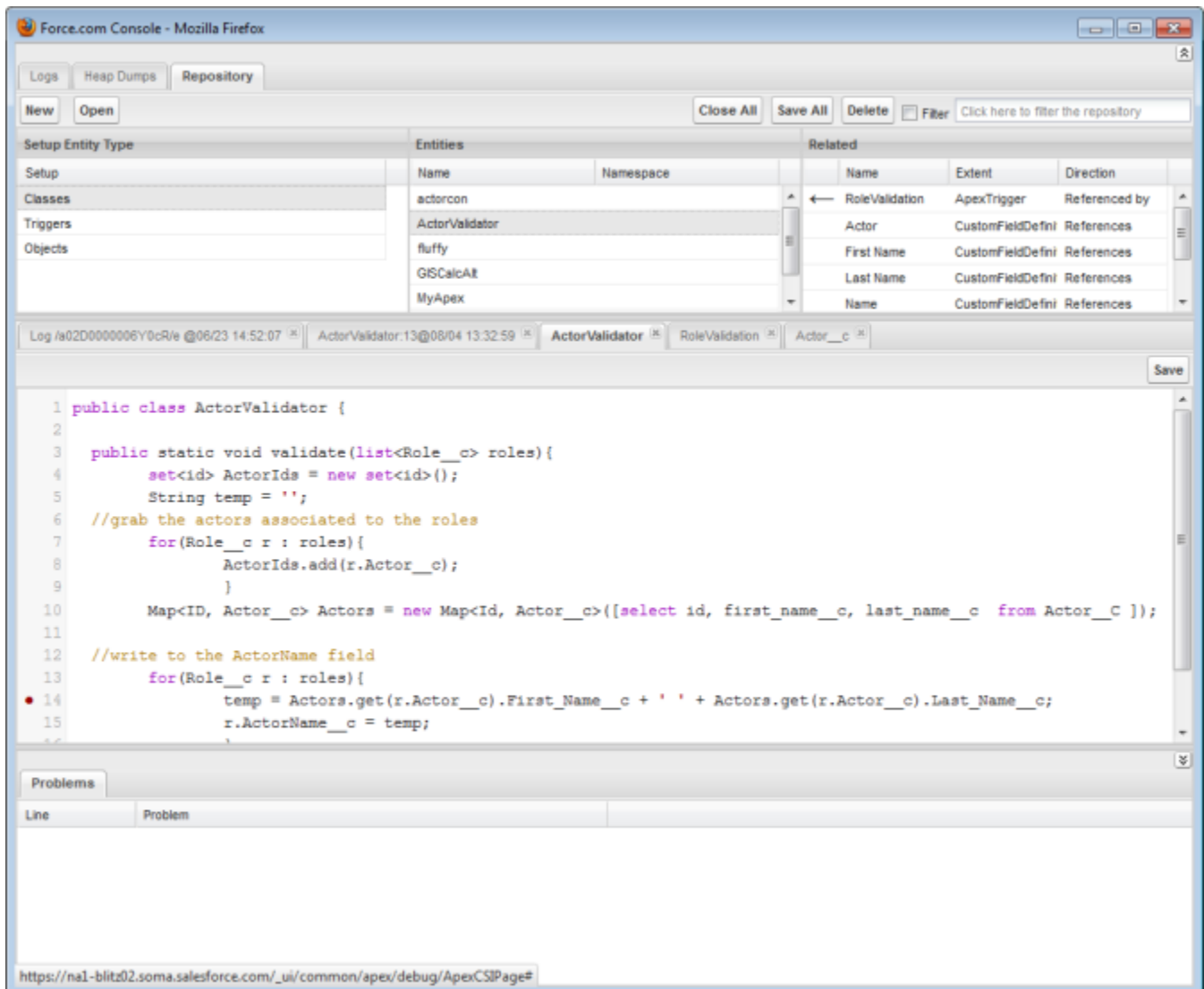
The Developer Console has a number of panels that let you investigate the performance characteristics of your apps. You can open a debug log and review the Executed Units tab, which breaks up the request both by time and type. This tab further categorizes the timings by methods, queries, workflows, callouts, DML, validations, and triggers, which gives you a clear idea of where to find performance issues.

Developer Console Tools

Tools in the new Developer Console include:

- The debug logs browser and System Log view you are already familiar with
- A new Heap Dump Inspector
- A new Source Code Editor
- A new Source Repository browser
- A new read-only object schema viewer

You can have all of them available at once, in a single window.



There are three tabbed navigation panels used for browsing to things you can examine:

- **Logs**—allows you to open existing debug logs and, when open, allows you to capture new logs from actions you take in your organization.
- **Heap Dumps**—allows you to view a snapshot of the state of execution—objects and variables, references to and from instantiated objects—with links to the related source code.
- **Repository**—allows you to navigate to and open your objects, Apex classes and triggers, etc.

The Developer Console offers four detail views that allow you to investigate, filter, and in the case of source code, edit the entities in your app:

- The **System Log** view can be considered a context-sensitive execution viewer, showing the source of an operation, what triggered that operation, and what occurred afterward. Use this view to inspect debug logs that include database events, Apex processing, workflow, and validation logic.
- The **Heap Dump Inspector** view allows you to browse snapshots of the state of objects in memory at the time of the capture, including references between objects, and the ability to view variables in more detail than offered in the System Log view, including individual items in collections. Use the Heap Dump Inspector to investigate what objects are in memory at a specific point of execution and why, that is, what other objects hold references to them.

- The Source Code view displays Apex source files and allows you to edit them in the Developer Console. Using the Repository browser, you can open a set of related source code files and switch between them more quickly than using the Setup sidebar.
- The Schema Quick View allows you to see the fields of a custom object and their data types.

For more information, see [About the Force.com Developer Console](#).

See Also:

[Database.com Console Enhancements](#)

Recycle Bin Now Stores Records for 15 Days

In Winter '12, the recycle bin now stores records for 15 days before those records are permanently deleted. Previously, the recycle bin stored records for 30 days.

See Also:

[Database.com Console Enhancements](#)

Unique Subdomain for the Database.com Console

In Winter '12, subdomains are automatically generated for each organization. The reason for this change is to provide a more effective abstraction of instance details from URL structures, and a more consistent experience for administrators and end users.

All existing HTTP links to Database.com Console pages are automatically forwarded to equivalent pages that include the new subdomain. You should update your bookmarks to reflect the new URL. Login URLs for the Database.com Console and API remain the same.

If desired, you can create a custom subdomain for your organization by using the My Domains feature. For more information, see “My Domain Overview” in the Salesforce online help

See Also:

[Database.com Console Enhancements](#)

Permission Sets—Generally Available



You asked for it! This enhancement is from an [idea](#) on IdeaExchange.

In Winter '12, permission sets are generally available. A permission set is a collection of settings and permissions that give users access to various tools and functions. The settings and permissions in permission sets are also found in profiles, but permission sets extend users' functional access without changing their profiles. For example, to give users access to a custom object, create a permission set, enable the required permissions for the object, and assign the permission set to the users. You never have to change profiles, or create a profile for a single use case. While users can have only one profile, they can have multiple permission sets.

Instead of creating “one-off” profiles, you can use permission sets to grant additional permissions or other access to a user. Permission sets can represent a concept, like a user's job title, or more specific access settings, such as a set of tasks that a user needs to do, like deleting or transferring records.

System requirements that previously existed only on the profile can now be handled by assigning permission sets to users. For example, before permission sets, only users with “View Setup and Configuration” enabled on their profile could be assigned to a delegated administrator group. With permission sets, users can be assigned to a delegated administrator group if “View Setup and Configuration” is enabled on either their profile or any of their assigned permission sets. Conversely, administrators can't reassign a delegated administrator's profile or remove permission sets that would result in a user no longer having “View Setup and Configuration.”

See Also:

- [Overview of User Permissions and Access](#)
- [Creating Permission Sets](#)
- [Using Permission Set Lists](#)
- [Working in a Permission Set's Overview Page](#)
- [About App and System Permissions](#)
- [Assigning Permission Sets](#)
- [Searching Permission Sets](#)
- [Using a Permission Set's Assigned Users Page](#)
- [Revoking Permissions and Access](#)

Overview of User Permissions and Access

User permissions and access settings specify what users can do within an organization. For example, permissions determine a user's ability to edit an object record, view the Setup menu, empty the organizational recycle bin, or reset a user's password. Access settings determine other functions, such as access to Apex classes, app visibility, and the hours when users can log in.

Permissions and access settings are specified in user profiles and permission sets. Every user is assigned only one profile, but can also have multiple permission sets.

When determining access for your users, it's a good idea to use profiles to assign the minimum permissions and access settings for specific groups of users, then use permission sets to grant additional permissions.

Because you can assign many permission sets to users and permission sets are reusable, you can distribute access among more logical groupings of users, regardless of their primary job function. For example, you could create a permission set that gives read access to a custom object and assign it to a large group of users, and create another permission set that gives edit access to the object and assign it to only a few users. You can assign these permission sets to various types of users, regardless of their profiles.

The following table shows the types of permissions and access settings that are specified in profiles and permission sets. Some profile settings aren't included in permission sets.

Permission or Setting Type	In Profiles?	In Permission Sets?
Object permissions	✓	✓
Field permissions	✓	✓
User permissions (app and system)	✓	✓
Apex class access	✓	✓
Service provider access (if Database.com is enabled as an identity provider)	✓	✓
Login hours	✓	
Login IP ranges	✓	

See Also:

[Permission Sets—Generally Available](#)

Creating Permission Sets

User Permissions Needed	
To create permission sets:	“Manage Users”

You can either clone an existing permission set or create a new one. A cloned permission set starts with the same user license and enabled permissions as the permission set it is cloned from, while a new permission set starts with no user license selected and no permissions enabled. You can create up to 1,000 permission sets.

1. Click **Manage Users > Permission Sets**.
2. Do one of the following:
 - To create a permission set with no permissions enabled, click **New**.
 - To create a permission set based on an existing set, click **Clone** next to the set you want to copy. You can also select the permission set and click **Clone** in the overview page or one of the settings pages.



Note: Clone a permission set only if the new one will have the same user license as the original. In a cloned permission set, you can't select a different license.

3. Enter a label, API name, and description.

The API name is a unique name used by the Force.com API. It must begin with a letter, and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.

4. If this is a new permission set, select the user license that matches the users who will use this permission set. For example, if you plan to assign this permission set to users with the Database.com User license, select Database.com User.
5. Click **Save**.

The permission set overview page appears. From here you can navigate to the permissions you want to add or change.


See Also:

[Permission Sets—Generally Available](#)

Using Permission Set Lists

User Permissions Needed	
To view permission sets, and print permission set lists:	“View Setup and Configuration”
To delete permission sets and permission set list views:	“Manage Users”

To view the permission sets in your organization, click **Manage Users > Permission Sets**. In the permission sets list page, you can:

- Show a filtered list of permission sets by selecting a view from the drop-down list
- [Create a list view or edit an existing view](#)
- Delete a list view by selecting it from the drop-down list and clicking **Delete**
- [Create or clone a permission set](#)
- Print a list view
- Refresh the list view by clicking 
- [Edit permissions directly in a list view](#)
- [View or edit a permission set](#)
- If it's not assigned to any users, remove a permission set by clicking **Del**

See Also:

[Permission Sets—Generally Available](#)


[Creating and Editing Permission Set List Views](#)

[Editing Permission Sets from a List View](#)

Creating and Editing Permission Set List Views

User Permissions Needed	
To create, edit, and delete permission set list views:	“Manage Users”

You can create and edit permission set list views to show a list of permission sets with specific fields and permissions. For example, you could create a list view of all permission sets in which “Modify All Data” is enabled.

1. In the Permission Sets page, click **Create New View**, or select a view and click **Edit**.
2. Enter the view name.
3. Under Specify Filter Criteria, specify the conditions that the list items must match, such as `Modify All Data equals True`.
 - a. Type a setting name, or click the lookup icon  to search for and select the setting you want.
 - b. Choose a filter operator.
 - c. Enter the value that you want to match.
 - d. To specify another filter condition, click **Add Row**. You can specify up to 25 filter condition rows.
To remove a filter condition row and clear its values, click **X**.
4. Under Select Columns to Display, specify the settings that you want to appear as columns in the list view. You can add up to 15 columns.
 - a. From the Search drop-down list, select a setting type.
 - b. Enter part or all of a word in the setting you want to add and click **Find**.
 Note: If the search finds more than 500 values, no results appear. Refine your search criteria to show fewer results.
 - c. To add or remove columns, select one or more column names and click the **Add** or **Remove** arrow.
 - d. Use the **Top**, **Up**, **Down**, and **Bottom** arrows to arrange the columns in the sequence you want.
5. Click **Save**, or if you're cloning an existing view, rename it and click **Save As**.

See Also:

[Using Permission Set Lists](#)

Editing Permission Sets from a List View

User Permissions Needed	
To edit multiple permission sets from the list view:	“Manage Users”
	AND
	“Customize Application”
	AND
	“Mass Edits from Lists”

You can change permissions in up to 200 permission sets directly from the list view, without accessing individual permission sets.



Note: Use care when editing permission sets with this method. Making mass changes may have a widespread effect on users in your organization.

1. Select or create a list view that includes the permission sets and permissions you want to edit. Editable fields display a pencil icon (✎) when you hover over the field, while non-editable fields display a lock icon (🔒).
2. To edit multiple permission sets, select the checkbox next to each one you want to edit. If you select permission sets on multiple pages, the selections on each page are remembered.
3. Double-click the permission you want to edit. For multiple permission sets, double-click the permission in any of the selected permission sets.
4. In the dialog box that appears, enable or disable the permission. In some cases, changing a permission may also change other permissions.
5. To change multiple permission sets, select **All n selected records** (where n is the number of permission sets you selected).
6. Click **Save**.

If you edit multiple permission sets, only those that support the permission you are changing will change. For example, if you use inline editing to enable “Modify All Data” in a permission set, but because of its user license the permission set doesn't have “Modify All Data,” the permission set won't change.

If any errors occur, the error message lists each permission set and a description of the error. Click the permission set name to open its overview page. The permission sets you've clicked appear in the error window in gray, strike-through text.



Note: To view the error console, pop-up blockers must be disabled for the Database.com domain. To check if your browser allows pop-up windows, click **My Personal Information > Reminders**, and then click **Preview Reminder Alert**.

Any changes you make are recorded in the setup audit trail.

See Also:

[Using Permission Set Lists](#)

Working in a Permission Set's Overview Page

User Permissions Needed	
To view permission sets:	“View Setup and Configuration”
To delete permission sets and edit permission set properties:	“Manage Users”

A permission set's overview page provides an entry point for all of the permissions in a permission set. To open a permission set overview page, click **Manage Users** > **Permission Sets** and select the permission set you want to view.

From the permission set overview page, you can:

- [Search for an object, setting, or permission](#)
- [Create a permission set based on the current permission set](#)
- If it's not assigned to any users, remove the permission set by clicking **Delete**
- Change the permission set label, API name, or description by clicking **Edit Properties**
- [View a list of users who are assigned to the permission set](#)
- View or edit:
 - ◇ Object and field permissions
 - ◇ App permissions
 - ◇ Apex class access settings
 - ◇ System permissions
 - ◇ Service providers (if you've enabled Database.com as an identity provider)

See Also:

[Permission Sets—Generally Available](#)

About App and System Permissions

In permission sets, permissions and settings are organized into app and system categories, which reflect the rights users need to administer and use system and app resources.

App Permissions

The Apps section of the permission sets overview page contains settings that are directly associated with the business processes that the apps enable. The Apps section contains links to:

- Object and field permissions
- App permissions
- Apex class access

System Permissions

In the permission sets overview page, the System section contains links to:

- System permissions

- Service providers (if you've enabled Database.com as an identity provider)

See Also:

[Permission Sets—Generally Available](#)

Assigning Permission Sets

User Permissions Needed	
To view users that are assigned to a permission set:	“View Setup and Configuration”
To assign permission sets:	“Manage Users”

From the user detail page, you can assign permission sets or remove a permission set assignment.

1. Click **Manage Users > Users**.
2. Select a user.
3. In the Permission Set Assignments related list, click **Edit Assignments**.
4. To assign a permission set, select it from the Available Permission Sets box and click **Add**. To remove a permission set assignment, select it from the Enabled Permission Sets box and click **Remove**.



Note: The Permission Set Assignments page shows only permissions sets that match the user's license.

5. Click **Save**.

See Also:

[Permission Sets—Generally Available](#)

Searching Permission Sets

User Permissions Needed	
To search permission sets:	“View Setup and Configuration”

On any of the detail pages, type at least three consecutive letters of an object, setting, or permission name in the **Find Settings...** box. The search terms aren't case-sensitive. As you type, suggestions for results that match your search terms appear in a list. Click an item in the list to go to its settings page. You can search for:

Item	Example
Objects	Say your application contains an Albums object. Type <code>albu</code> in the Find Settings box, then select Albums.
Fields (Find the object that the field is part of.)	Say your Albums object contains a Description field. To find the Description field for albums, type <code>albu</code> , select Albums, and scroll down to Description under Field Permissions.
App permissions	Type <code>lEaD</code> , then select Import Leads.
Apex class access settings	Type <code>apex</code> , then select Apex Class Access.
System permissions	Type <code>api</code> , then select API Enabled.
Service providers (available only if Database.com is enabled as an identity provider)	Type <code>serv</code> , then select Service Providers.

If no results appear in a search:

- The permission, object, or setting you're searching for may not be available in the current organization.
- The item you're searching for may not be available for the user license that's associated with the current permission set.
- Be sure your search terms have at least three consecutive characters that match the object, setting, or permission name.
- Be sure the search term is spelled correctly.

See Also:

[Permission Sets—Generally Available](#)

Using a Permission Set's Assigned Users Page

User Permissions Needed	
To view users that are assigned to a permission set:	“View Setup and Configuration”

To view all users that are assigned to a permission set, from the permission set overview page, click **Assigned Users**. From the assigned users page, you can:

- Create one or multiple users
- Reset passwords for selected users
- Edit a user
- View a user's detail page by clicking the name, alias, or username
- View or edit a profile by clicking the profile name

See Also:

[Permission Sets—Generally Available](#)

Revoking Permissions and Access

You can use profiles and permission sets to grant access, but not to deny access. Any permission granted from either a profile or permission set is honored. For example, if “Transfer Record” isn't enabled in Jane Smith's profile, but is enabled in two of her permission sets, she can transfer records regardless of whether she owns them. To revoke a permission, you must remove all instances of the permission from the user. You can do this with the following actions—each has possible consequences.

Action	Consequence
Disable a permission or remove an access setting in the profile and any permission sets that are assigned to the user.	The permission or access setting is disabled for all other users assigned to the profile or permission sets.
If a permission or access setting is enabled in the user's profile, assign a different profile to the user.	The user may lose other permissions or access settings associated with the profile or permission sets.
AND If the permission or access setting is enabled in any permission sets that are assigned to the user, remove the permission set assignments from the user.	

To resolve the consequence in either case, consider all possible options. For example, you can clone the assigned profile or any assigned permission sets where the permission or access setting is enabled, disable the permission or access setting, and assign the cloned profile or permission sets to the user. Another option is to create a base profile with the least number of permissions and settings that represents the largest number of users possible, then create permission sets that layer additional access.

See Also:

[Permission Sets—Generally Available](#)

Security Enhancements

Winter '12 introduces several new features that enhance security.

Updated Encrypted Custom Field Permission

You don't have to contact salesforce.com to turn on this permission. It is now on for all organizations.

Revoking Remote Access Tokens Using OAuth API

You can now use the OAuth API to revoke or refresh access tokens and all related access tokens. For example, developers may use this feature when configuring a Log Out button in their application.

New Chatter API Endpoint URLs For Identity Service

The following Chatter URLs have been added to the Identity Service.

- `feeds`
- `feed-items`
- `groups`
- `users`

New `scope` Parameter in OAuth

You can use the `scope` parameter to fine-tune what a client application can access in your Database.com organization. Scopes will be available after all Salesforce instances are upgraded to Winter '12. Go to trust.salesforce.com for exact dates.

Single Sign-On Changes

If you change your Single Sign-On settings, your sandbox Single Sign-On can be disrupted. The login history entries read "SAML Idp Initiated SSO - Failed: Recipient Mismatched." This is caused by differences between the Identity Provider's ACS URL configuration and the value of the Salesforce Login URL on the Single Sign-On Settings page.

Beginning in Spring '12, whenever you save the Single Sign-On Settings page in a sandbox organization, the Salesforce Login URL value is reformatted. This will happen even if you made no changes to the page. Be sure to review the Salesforce Login URL value any time you've saved the Single Sign-On Settings page. If necessary, update your Identity Provider's ACS URL configuration (the `Recipient` attribute of the `Response` element in the SAML login request XML document) with the updated Salesforce Login URL value from the Single Sign-On Settings page. The page is at **Administration Setup > Security Controls > Single Sign-On Settings**.

Apex Enhancements

Apex includes the following enhancements in Winter '12. Refer to the [Database.com Apex Code Developer's Guide](#) for complete information about Apex.

The following enhancements are generally available:

Apex REST

The Apex REST API was previously available as a pilot in Summer '11. You can use Apex REST to implement custom Web services in Apex and expose them through the REST architecture.

Six new annotations have been added to Apex for Apex REST: `@RestResource`, `@HttpDelete`, `@HttpGet`, `@HttpPatch`, `@HttpPost`, and `@HttpPut`. Define your Apex class with the `@RestResource` annotation to expose it as a REST resource. Similarly, add annotations to your methods to expose them through REST. For example, you can add the `@HttpGet` annotation to your method to expose it as a REST resource that can be called by an HTTP GET request.

Cookie support has been removed from `RestRequest` and `RestResponse`. The `addCookie` method and `cookies` property have been removed from both of these classes, and use of the following headers in the `addHeader` method isn't allowed: "cookie", "set-cookie", "set-cookie2", "content-length" and "authorization".

For more information about Apex REST, see the [Database.com Apex Code Developer's Guide](#).

API for Asynchronous Test Runs—Beta

Two API objects are now available to enable starting asynchronous test runs and checking test results:

- `ApexTestQueueItem`: Represents a single Apex class in the Apex job queue.
- `ApexTestResult`: Represents the result of an Apex test method execution.

Common Test Data Creation Methods in Public Test Classes

Classes annotated with `@isTest` no longer have to be private. You can now create public test classes that expose common methods for test data creation. These public methods can be called by tests outside the test class for setting up data that the tests need to run against.

Methods of a public test class can only be called from a running test, that is, a test method or code invoked by a test method, and can't be called by a non-test request.

This is an example of a public test class that contains utility methods for test data creation:

```
@isTest
public class TestUtil {

    public static void createTestData() {
        // Create some test invoices
    }

}
```

Higher Governor Limits in Some Categories

Limits for the heap and HTTP callout request and response sizes have doubled:

- Total heap size is 6 MB.
- Maximum Email Services heap size is 36 MB.
- The maximum size of a callout request or response for an HTTP request or Web services call is 6 MB.

In addition, some governor limits are higher for Batch Apex and future methods:

- Total number of SOQL queries issued is 200 instead of 100.
- Total number of executed code statements is 1,000,000 instead of 200,000.
- Total heap size for Batch Apex and future methods is 12 MB instead of 6 MB.

Method Size Limit

The size of a method is limited to 65,535 bytecode instructions in compiled form. Large methods that exceed the allowed limit cause an exception to be thrown during the execution of your code.

JSON Support

JavaScript Object Notation (JSON) support has been added. Using JSON classes, you can now parse JSON content and serialize Apex objects into the JSON format. The following classes have been added for JSON support:

- `System.JSON`: Contains methods for serializing Apex objects into JSON format and deserializing JSON content that was serialized using the `serialize` method in this class.
- `System.JSONGenerator`: Contains methods used to serialize Apex objects into JSON content using the standard JSON encoding.
- `System.JSONParser`: Represents a parser for JSON-encoded content.

New Schema Methods

The following new method has been added to the `Schema.ChildRelationship` class:

- `isRestrictedDelete`: Returns `true` if the parent object can't be deleted because it is referenced by a child object, `false` otherwise. This prevents users from deleting any records that are part of a lookup relationship.

The following new methods have been added to the `Schema.DescribeFieldResult` class:

- `isCascadeDelete`: Returns `true` if the child object is deleted when the parent object is deleted, `false` otherwise. This allows a cascade delete when a parent lookup record is deleted, allowing users to delete large numbers of records without regard for sharing or visibility constraints.
- `isRestrictedDelete`: Returns `true` if the parent object can't be deleted because it is referenced by a child object, `false` otherwise. This prevents users from deleting any records that are part of a lookup relationship.

New System Methods: `isBatch`, `isFuture`, and `isScheduled`

The following methods have been added to the `System` class:

- `isBatch`: Determines if the currently executing code is invoked by a batch Apex job. Since a future method can't be invoked from a batch Apex job, use this method to check if the currently executing code is a batch Apex job before you invoke a future method.
- `isFuture`: Determines if the currently executing code is invoked by code contained in a method annotated with `@future`. Since a future method can't be invoked from another future method, use this method to check if the current code is executing within the context of a future method before you invoke a future method.
- `isScheduled`: Determines if the currently executing code is invoked by a scheduled Apex job.

ReadOnly Annotation—Generally Available

The `@ReadOnly` annotation was previously released as a pilot in Spring '11 and is now generally available in Winter '12.

The `@ReadOnly` annotation allows you to perform unrestricted queries against the database. All other limits still apply. It's important to note that this annotation, while removing the limit of the number of returned rows for a request, blocks you from

performing the following operations within the request: DML operations, calls to `System.schedule`, and calls to methods annotated with `@future`.

The `@ReadOnly` annotation is available for Web services and the `Schedulable` interface. To use the `@ReadOnly` annotation, the top level request must be in the schedule execution or the Web service invocation.

Additional Apex Enhancements

Clone Methods Clear IDs By Default

The optional `opt_preserve_id` Boolean argument of the `List.clone` and the `sObject.clone` methods now defaults to `false` if not set. This means that by default the ID fields are cleared in the cloned copies.

For Apex saved using Salesforce API version 22.0 or earlier, the default value for the `opt_preserve_id` argument is `true`, that is, the IDs are preserved.

Debug Log Download Link in the Apex Test Result Page

The Apex Test Result page now contains a link that allows you to download the debug log that is generated after running test methods of Apex classes using **Run Test** and **Run All Tests**. To get a copy of the debug log, click **Download**. The Apex Test Result page no longer displays the debug log output. The Apex Test Result page is displayed after you run all tests or just the tests of a single class. To run all tests, click **Develop > Apex Classes** to access the Apex Classes page, and then click **Run All Tests**. To run the tests of a single class, click **Your Class Name > Run Test** from the Apex Classes page.

Getting the Currency Code for Single Currency Organizations



You asked for it! This enhancement is from [this idea](#) on the IdeaExchange.

The `UserInfo.getDefaultCurrency` method now returns the organization's currency code for single currency organizations.

For Apex saved using Salesforce API version 22.0 or earlier, `getDefaultCurrency` returns `null` for single currency organizations.

String Conversion of Number Fields

Previously, when `String.valueOf` was called with a field of type `Number` of an `sObject`, it incorrectly treated the number field as a `Decimal` when converting it to a `String` and used the `String.valueOf(Decimal d)` method to perform the conversion to a `String`. Apex now correctly converts a number field to a `Double` before performing the conversion and uses the corresponding `String.valueOf(Double d)` method to convert the `Double` value to a `String`. One side effect of this change is that converted `String` values of number fields that have no decimal fraction now have a decimal point (`.0`) in them where they didn't before.

Column Number in Runtime Errors

Column numbers in runtime errors, including exception error messages, are always 1. Prior to Winter '12, column numbers in runtime errors were the first character of the statement where the error occurred.

Changes in Code Coverage

- Apex doesn't track the execution time per code location or the number of executions of a code location. Apex only tracks these metrics in the cumulative profiling section of the debug log in these profiling categories: SOSL operations, SOQL operations, DML operations, and method invocations.
- The first column number of code is always reported as 1. Subsequent column numbers on the same line continue to be reported as they used to in prior releases, namely the column number of the first character of the code statement.

Fix for Uncaught Exception Bug for Apex Code Saved Using Salesforce API version 20.0 or earlier

Exceptions were not being caught in the catch block under a certain condition for Apex code saved using Salesforce API version 20.0 or earlier. The exception wasn't being caught in the catch block if the try block also threw an exception. This bug has now been fixed for all versions of Apex.

Debug Log Formatting Changes

Some debug log information has new formatting in the debug log. For details, see [What's New?](#) in the *[Database.com Apex Code Developer's Guide](#)*.

API Enhancements

Winter '12 (API version 23.0) improvements:

- [New and Changed Objects](#), including [Chatter API Objects](#)
- [Metadata API Enhancements](#)
- [REST API Enhancements](#)
- [Streaming API—Pilot](#)
- [Chatter REST API Enhancements](#)
- [New and Changed SOAP Calls](#)

New and Changed Objects

Generally Available Enhancements

New Objects

The following objects have been added in API version 23.0:

- The following permission set objects were added in API version 22.0 as part of a pilot release. They are generally available in API version 23.0:

Object	Description
PermissionSet	Represents a set of permissions that's used to grant additional permissions to one or more users without changing their profile.
PermissionSetAssignment	Represents the association between a User and a PermissionSet.

- The following objects have been added so that administrators with the “Manage Chatter Messages” permission can view all users' Chatter messages, for example for compliance purposes:

Object	Description
ChatterConversation	Represents a private conversation in Chatter, consisting of messages that conversation members have sent or received.
ChatterConversationMember	Represents a member of a private conversation in Chatter. A member has either sent messages to or received messages from other conversation participants.
ChatterMessage	Represents a message sent as part of a private conversation in Chatter.

- The ChatterActivity object was added to represent the number of posts and comments made by a user and the number of comments and likes on posts received by the same user.
- The following objects have been added for asynchronous execution of Apex test methods.



Note: The API for asynchronous test runs is a Beta release.

Object	Description
ApexTestQueueItem	Represents a single Apex class in the Apex job queue.
ApexTestResult	Represents the result of an Apex test method execution.

Changed Objects

The following objects have been changed in API version 23.0:

- Users with “View All Documents” permission can now directly query `FeedComment` and `FeedItem`. For `FeedItem`, you no longer need an ID filter. For `FeedItem`, you can just query for all feed items.
- The `FeedLike` object now includes a new field to differentiate between likes on posts and likes on comments:
 - ◊ `FeedEntityId`, which contains the ID of a feed item or comment a user liked. If the user liked a comment, `FeedEntityId` is set to the ID of the comment. If the user liked a feed item, `FeedEntityId` is set to the ID of the feed item. `FeedEntityId` is an optional field. The default value is the ID of the feed item.

See Also:

[Chatter API Objects](#)

Chatter API Objects

New Chatter Objects

The following new objects have been added in API version 23.0.

Object	Description
<code>ChatterActivity</code>	<code>ChatterActivity</code> represents the number of posts and comments made by a user and the number of comments and likes on posts received by the same user.
<code>ChatterConversation</code>	Represents a private conversation in Chatter, consisting of messages that conversation members have sent or received.
<code>ChatterConversationMember</code>	Represents a member of a private conversation in Chatter. A member has either sent messages to or received messages from other conversation participants.
<code>ChatterMessage</code>	Represents a message sent as part of a private conversation in Chatter.

Changed Chatter Objects

The following objects have been changed in API version 23.0.

- Both the `FeedComment` and `FeedItem` objects can be queried directly. Prior to API version 23.0, `FeedComment` could only be queried as a child of a feed (such as a record feed, `AccountFeed`, and so on) and `FeedItem` required an ID filter.
- The `Type` field of the `FeedItem` object now includes new values:
 - ◊ `DashboardComponentSnapshot`—a posting of a dashboard snapshot on a feed.
 - ◊ `ApprovalPost`—automatically generated by a feed query on an approval item.

- The CollaborationGroup object now includes the `CanHaveGuests` field, which indicates if a private group allows customers. Chatter customers are users outside your company's email domains who can only see groups they're invited to and interact with members of those groups; they can't see any Salesforce information.


See Also:

[New and Changed Objects](#)

Metadata API Enhancements

New Metadata Types

The following metadata types are new in Metadata API version 23.0:

Metadata Type	Description
PermissionSet	<p>Generally available in Winter '12. Represents a set of permissions that's used to grant additional permissions to one or more users without changing their profile. You can use permission sets to grant access, but not to deny access.</p> <p> Note: A pilot version of this object was available in version 22.0.</p>

Updated Metadata

The following metadata fields have been added or changed in Metadata API version 23.0:

Metadata Type or Related Object	Field	Change	Description
Profile	fieldPermissions	New	This field replaces <code>fieldLevelSecurities</code> .
Profile (ProfileFieldLevelSecurity)	hidden	Removed	This field has been replaced by the <code>readable</code> field.
Profile (ProfileFieldLevelSecurity)	readable	New	Indicates whether this field is readable (<code>true</code>) or not (<code>false</code>). This field replaces the <code>hidden</code> field.

REST API Enhancements

The Force.com REST API leverages the simplified approach of REST to allow developers to more easily interact with other Web 2.0 applications such as Amazon.com AWS, Microsoft Azure, Google, Facebook, Twitter, and others:

- Interact with the application in a simplified way
- Integrate with third-party cloud systems and services
- Support mashup and Web 2.0 projects

If you need to move large amounts of data, use the Bulk API, which is built on RESTful principles, but is asynchronous and optimized for large data-loading tasks.

Apex REST

The Apex REST API was previously available as a pilot in Summer '11. You can use Apex REST to implement custom Web services in Apex and expose them through the REST architecture.

For more information, see [Apex Enhancements](#) on page 25.

Streaming API—Pilot

The Database.com Streaming API lets you push data from Database.com to clients in a secure and scalable way using a persistent HTTP connection.

For more information, see [Database.com Streaming API Developer's Guide](#).

API Version Required in URL

When you subscribe to a topic, you must now include the API version at the end of the URL.

- Previous Versions: `https://instance.salesforce.com/cometd`
- Current Version (23.0): `https://instance.salesforce.com/cometd/23.0`

Topics are Organized Under /topic/

Push topics are now organized under /topic. Therefore, when you subscribe to a channel, you must now include “/topic/” before the topic name.

- Previous Versions: `/TopicName`
- Current Version (23.0): `/topic/TopicName`

Authentication Header Uses OAuth

The HTTP authentication header is sent in each request. The authentication header now uses OAuth and not a cookie. This means the authentication header must contain the session ID as shown in the following example.

- Previous Versions: `Cookie: sid=sessionId`
- Current Version (23.0): `Authorization: OAuth sessionId`

Timeout Changes

Each Streaming API client logs into an instance and maintains a session. When the client handshakes, connects, or subscribes, the session timeout is restarted. In previous versions, the Streaming API client session timed out according to the timeout rules of the organization. For more information, see [Clients and Timeouts](#) in the [Database.com Streaming API Developer's Guide](#).

- Previous Versions: If your organization timeout was two hours of inactivity, the Streaming API client session would time out after two hours even if that client session had activity.
- Current Version (23.0): When the Streaming API client session handshakes, connects or subscribes, the session timeout is restarted.

Chatter REST API Enhancements

With Winter '12, Chatter API is now generally available. The Chatter API is a REST API that provides access to Chatter feeds and social data such as users, groups, followers, and files. It's used by developers who want to integrate Chatter into mobile applications, intranet sites, and third-party Web applications.



Attention: Version 22.0 was a pilot. Once your organization is upgraded to version 23.0, you can no longer access any version 22.0 resources.

Winter '12 (Chatter API version 23.0) improvements:

- [General Improvements](#)
- [New and Changed Resources](#)
- [New and Changed Response Bodies](#)

General Improvements

Separate Limits for Chatter API

The Chatter API request limit is 200 requests per OAuth consumer per user per hour. For example, if you have created three remote access applications (each of which represents a different OAuth consumer with its own `Consumer Key` and `Consumer Secret`), each user of those apps has 200 requests per hour. If a user issues more than 200 requests per hour, Chatter API returns a 503 Service Unavailable error code.



Note: Stated limits aren't a promise that the specified resource is available at its limit in all circumstances. For example, load, performance and other system issues might prevent some limits from being reached. The stated limit may change without notice. Applications should make efficient use of available requests and gracefully handle the 503 error code.

Use Rich Input When Adding Posts

HTTP POST methods take either a request body or request parameters. Use the request body to include rich input using serialized JSON or XML in the body of the request. If you pass in both a request body and a request parameter, a 400 error is thrown.

Specify Return as JSON or XML

You can now append either `.json` or `.xml` to the URI of any resource to specify how the return is formatted. For example, `/chatter/feeds/filter/me/001/feed-items.xml`.

Separate Chatter API Permissions

Chatter API now has its own permissions. You can have the Chatter API enabled and still disable other APIs. Contact your salesforce.com representative about organization permissions.

Encoded Returns

In order to make strings Web-safe, on output the response returns are encoded such that included string values are safe for use in a Web context. We escape the following characters in all strings:

Character	Escaped as
<	<

Character	Escaped as
>	>
"	"
'	'
\	\

Using these encodings lets the original characters be displayed as literals if the strings are inserted into a Web page, preventing them from being used to insert unexpected HTML or Javascript into the page.

If you don't want the Chatter API to encode these characters, set the `X-Chatter-Entity-Encoding` HTTP header in your request to `false`.

Chatter API does special encoding of any URL values included in response bodies. The main part of the URL is URL-encoded as per [RFC2396](#), and the query string is HTML-form encoded. This encoding cannot be turned off. URL values are indicated in the returns as having the type `URL`.

Post @mentions

You can now post @mentions in a feed. You must use the appropriate request body, such as comment input or feed-item input, with the message segment mention. The following is an example request body:

```
{ "body" :
  {
    "messageSegments" : [
      {
        "type": "mention",
        "id" : "005D0000001GpHp"
      },
      {
        "type": "text",
        "text" : " Did you see the new plans?"
      }
    ]
  }
}
```



Note: No spaces are added either before or after the @mention, so the text message segment includes a space.

New and Changed Resources

- The Users resource now provides access to a user's messages from private conversations in Chatter. You can use this resource to return Chatter messages for the current, logged in user.
- The Users resource now provides access to a user's Chatter recommendations. You can use this resource to return people, group, and record recommendations for the current, logged in user.
- The Users and the Groups resources now accept comma-separated lists of users or groups with the batch resource, so that you can use a single call to get information about one or more users or groups.
- Topics is a new resource that provides access to an organization's trending topics.
- The following feed resources now have a top-level resource that returns a URL to the feed items for that feed:

◇ /feeds/news

- ◇ /feeds/record
- ◇ /feeds/to
- ◇ /feeds/user-profile

- The following resources now have a top-level resource that returns a collection of information:

- ◇ /groups
- ◇ /users

- /chatter is a new resource that returns URLs to the top-level resources for all collections such as /groups.
- /feeds is a new resource that returns a list of all the feed URLs available in an organization.
- The following feeds are new:

- ◇ /company
- ◇ /files
- ◇ /filter
- ◇ /groups
- ◇ /people

- /records is a new resource that returns information about the specified record.
- The /feed-items resource can now be used without specifying a feed item. However, the new request parameter q is required when you don't specify a feed item. Use the new request parameter q to query all feed items for the specified text.
- All feeds now have the following request parameters:
 - ◇ pageSize—used to specify the number of items per page. Default value is 25.
 - ◇ sort—used to specify the order feed items are returned in. Default is by created date, descending.

New and Changed Response Bodies

- The User Detail response body now includes the following:
 - ◇ Chatter Activity—a new response body that includes:
 - Total number of comments in the organization made by the user
 - Total number of comments in the organization received by the user
 - Total number of likes on posts (not on comments) in the organization received by the user
 - Total number of posts in the organization made by the user
 - ◇ companyName—name of the company.
 - ◇ Following Counts—a new response body that includes the number of people the current user is following, the number of records, as well as the total.
 - ◇ groupCount—number of groups the current user is following.
 - ◇ isChatterGuest—Boolean indicating if the current user is a Chatter customer or not.
 - ◇ mySubscription—either a Reference response body or null if the current user is not following the specified user.
 - ◇ type—returns the string User.
- The following are new response bodies from the Users resource related to sending and retrieving Chatter messages:
 - ◇ Conversation Detail—all the messages in a conversation.
 - ◇ Conversation Summary—the most recent message in a conversation.
 - ◇ Conversation Summary Collection—the conversations with the most recent messages.
 - ◇ Conversation Unread Count—the number of conversations that have unread messages.

- ◊ Message—the text of the message, timestamp, and information about the sender, recipients, and conversation that the message is part of.
- ◊ Message Body—the text of the message.
- ◊ Message Collection—the most recent messages.
- The following are new response bodies from the Users resource related to recommendations:
 - ◊ Explanation Summary—a brief recommendation explanation.
 - ◊ Recommendation—the recommended object, the action to take on that object, and a recommendation explanation.
- The following are new response bodies from the Users and Groups resource related to batch results:
 - ◊ Batch Results—a collection of individual Batch Results returns.
 - ◊ Batch Result Item—an individual batch result.
- Topics is a new response body that includes the names of the trending topics in an organization.
- Directory is a new response body that includes URLs for collection resources, such as `/groups` or `/feeds`.
- Feed Item response body now includes the following:
 - ◊ `currentUserLike`—a Reference response body or null.
 - ◊ `isLikedByCurrentUser`—boolean indicating if the feed item is liked by the current user.
 - ◊ `attachment`—may be null or one of the following new response bodies, depending on type:
 - Feed Item Attachment: Content
 - Feed Item Attachment: Link
 - ◊ `actor`—replaces user
- The Group response body now includes the following:
 - ◊ `canHaveChatterGuests`—boolean indicating if the group is an external group.
 - ◊ `myRole`—role of the current user in the group, such as owner, member, and so on.
 - ◊ `mySubscription`—a Reference response body if the current user is a member of this group, or null.
 - ◊ `type`—returns the string `CollaborationGroup`.
- The Group Page and Subscription Page response bodies now include the following:
 - ◊ `currentPageUrl`
 - ◊ `groups`—a collection of Group response bodies
 - ◊ `nextPageUrl`
 - ◊ `previousPageUrl`
- The Group Member response body now includes a Reference response body.
- The Group Member Page response body now includes `myMembership`—a Reference response body if the current user is a member of this group, or null.
- The Like Page response body now includes `myLike`—a Reference response body if the user likes the current like, null otherwise.
- Reference is a new response body that contains an ID and a URL link to the object's resource, such as `/user`.
- User Page is a new response body with the following information:
 - ◊ `currentPageUrl`
 - ◊ `nextPageUrl`
 - ◊ `previousPageUrl`
 - ◊ `users`—a collection of User Summary response body

- The User Summary response body now includes the following:
 - ◇ `companyName`—name of the company.
 - ◇ `isChatterGuest`—boolean indicating if the user is a Chatter customer.
 - ◇ `mySubscription`—a Reference response body if the current user is a member of this group, or null.
 - ◇ `type`—returns the string `User`.

New and Changed SOAP Calls

Calls are available using any Force.com API unless otherwise noted. Review the changes to existing calls in this release. You can also find links to changes in previous versions of objects.

Generally Available Enhancements

Changed Calls

The following calls have been changed in API version 23.0:

- The `getDeleted()` call now returns results for no more than 15 days prior to the day the call is executed (or earlier if an administrator purges the recycle bin). Previously, this call returned results for no more than 30 days prior to the call being executed.
- The `SendEmailResult` object returned by the `sendEmail()` call reflects a change in how we process mass email versus single email. Previously, a `success = true` result meant that the email was successfully accepted for delivery by the message transfer agent. There was no distinction between mass email and single email. Now, when sending mass email, a `success = true` result means that the email was successfully added to the queue for asynchronous processing. Even if the email was added to the queue, there can still be processing errors that prevent delivery to the intended recipients.

Database.com Java SDK—Beta

The Database.com Java SDK enables you to quickly build Java applications that access data stored in Database.com.

With this SDK, you can:

- Use Database.com to store your application data
- Interact with data already written to Database.com
- Leverage the Database.com user security model to manage data access for your application
- Extend existing Force.com applications with Java logic

The SDK is not tied to any particular runtime. You can build applications that run on your own server, on Amazon's EC2 or other cloud runtimes as long as you can make HTTPS connections to Database.com.

You can use the Database.com Java SDK with any organization that has Force.com API calls enabled.

The SDK consists of the following components.

- **API Connector**—Access existing Force.com APIs
- **JPA Provider**—Define entities in Java using the JPA standard and use Database.com as the persistent data store
- **OAuth authentication**—Leverage the security model in Database.com using OAuth2
- **Spring security plugin**—Use Spring Security for authentication and authorization in your applications
- **Spring MVC project template**—Create new projects quickly using a Maven archetype

For more details, see forcedotcom.github.com/java-sdk/force-sdk-overview.

Additional Database.com Enhancements

Recycle Bin Now Stores Records for 15 Days

In Winter '12, the recycle bin now stores records for 15 days before those records are permanently deleted. Previously, the recycle bin stored records for 30 days.

Database.com Light User License

In Winter '12, a new user license is available in Database.com. Database.com Light User license is designed for users who need only API access to data, need to belong to Chatter groups (but no other groups), and don't need to belong to roles or queues. Access to data is determined by organization-wide sharing defaults. To view your organization's active user licenses, click **Company Profile** > **Company Information**.

Domain Name Flexibility

After deployment, you now have the opportunity to change your domain name one time only.

Help and Training Enhancements

Videos

Like to learn by watching? Check out [Introducing Database.com](#) (4:12 minutes) and learn how to create:

- A Database.com schema
- Formula fields
- Lookup fields
- Validation rules

Workbooks

The following new or updated workbooks are now available:

- [Database.com Workbook](#)—Introduces you to Database.com via a series of tutorials. You'll create a database, build objects, add object relationships, create rules, and configure data sharing settings to control access.

Developer Guides

The following new or updated developer documents are now available:

- [Data Loader Developer's Guide](#)—A quick start for the Data Loader command-line interface has been added. This quick start shows you step-by-step how to import data by using the command line.
- [Object Reference](#)—Reference information about all standard and custom objects, data types, and related information.



Note: This information is also available in the [SOAP Web Services API Guide](#) as it always has been.

- [Database.com REST API Developer's Guide](#)—The REST API Developer's Guide updated for Database.com.
- [Database.com Apex Code Developer's Guide](#)—The Apex Code Developer's Guide updated for Database.com.
- [Database.com Chatter REST API Developer's Guide](#)—The Chatter REST API Developer's Guide updated for Database.com.
- [Database.com Streaming API Developer's Guide](#)—The Streaming API Developer's Guide updated for Database.com.
- [Database.com SOQL/SOSL Reference](#)—The SOQL/SOSL Reference updated for Database.com.
- [Database.com Bulk API Developer's Guide](#)—The Bulk API Developer's Guide updated for Database.com.