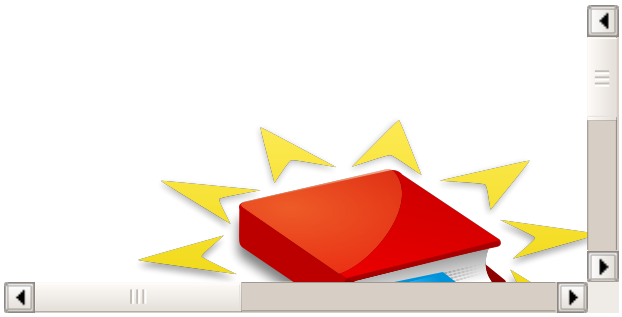
Zanata 1.6

Release Notes

Zanata Translation Management System

Edition 1



Joshua Wulf
jwulf@fedoraproject.org

Legal Notice

Abstract

New Features and Fixed Issues in the 1.6 release of Zanata

Table of Contents

Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the Liberation Fonts set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

`Mono-spaced Bold`

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keycaps and key combinations. For example:

> To see the contents of the file **`my_next_bestselling_novel`** in your current working directory, enter the **`cat my_next_bestselling_novel`** command at the shell prompt and press **`Enter`** to execute the command.

The above includes a file name, a shell command and a keycap, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from keycaps by the hyphen connecting each part of a key combination. For example:

> Press **`Enter`** to execute the command.
>
> Press **`Ctrl`**+**`Alt`**+**`F2`** to switch to the first virtual terminal. Press **`Ctrl`**+**`Alt`**+**`F1`** to return to your X-Windows session.

The first paragraph highlights the particular keycap to press. The second highlights two key combinations (each a set of three keycaps with each set pressed simultaneously).

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **`mono-spaced bold`**. For example:

> File-related classes include **`filesystem`** for file systems, **`file`** for files, and **`dir`** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

> Choose **System** → **Preferences** → **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).
>
> To insert a special character into a **gedit** file, choose **Applications** → **Accessories** → **Character Map** from the main menu bar. Next, choose **Search** → **Find...** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit** → **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

Mono-spaced Bold Italic or ***Proportional Bold Italic***

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

> To connect to a remote machine using ssh, type **ssh *username@domain.name*** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh john@example.com**.
>
> The **mount -o remount *file-system*** command remounts the named file system. For example, to remount the **/home** file system, the command is **mount -o remount /home**.
>
> To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

> Publican is a *DocBook* publishing system.

1.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }
}
```

1.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.

Note

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.

Warning

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.

2. We Need Feedback!

You should over ride this by creating your own local Feedback.xml file.

Chapter 1. Zanata 1.6

1.1. Zanata

Zanata is a web-based translation management system. Zanata provides an interface that allows a community of translators to work side-by-side on translation projects with nothing more than an Internet connection and a web-browser. Zanata provides APIs and client-side tools that allow software developers and documentation authors to push content to the server for translation and pull completed translations back for their project.

Zanata is an open source project sponsored by Red Hat. The project website is found at http://www.zanata.org. A hosted instance of Zanata that can be freely used to host and translate projects is found at http://translate.zanata.org.

1.2. Format of these Release Notes

These release notes cover Zanata 1.6, and comprises chapters covering:

- New Features added in the release
- Bugs that were resolved with the release

Each chapter is divided into three sections:

- Server Administration
- Project Owner
- Web Editor

Chapter 2. New Features

2.1. Administrator Features

2.1.1. Server statistics

The Zanata server is an application that runs on JBoss Application Server. The performance of a Zanata server is tied to the performance of its underlying JBoss Application Server. Without instrumenting that reports on statistics of the JBoss Application Server that hosts the Zanata server, it is difficult for a server administrator to monitor or tune the performance of a Zanata server.

JavaMelody, a Java EE server monitoring tool is now integrated with the Zanata server. Zanata server administrators can go to the "Administration" menu and select "Server Monitoring" to view the performance statistics for the Zanata instance.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=813161.

2.2. Project Owner Features

2.2.1. Allow Users to Create Projects

Creating a new translation project required server administrator privileges. Users had to request a translation project through a ticketing system and wait for a server administrator to create the project on their behalf.

Users can now create a translation project by following these steps:

1. Log in
2. Click “Project”
3. Click “Create project”
4. Fill in “Project ID”,”Name”,”Description”
5. Click “Save”

Note

The user who creates the project becomes the project's owner.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=727789.

2.2.2. Allow Project Owner to Edit All Translations in Project

In order to edit the translations for a language, a user must be a member of that language group in Zanata. Project owners were unable to edit the translations in their own project, unless they became a member of all language groups for which they had translations. This made applying an update to all translations in a project difficult.

Project owners can now edit all language translations in their project without joining multiple language groups.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=751673.

2.2.3. Maven pushTrans superseded by pushType

The maven **`-Dzanata.pushTrans`** option is now deprecated and is replaced by **`-Dzanata.pushType`**, which can take the following values:

source
: Push source documents only (This is the default value when not specified)

trans
Push translations only

both
Push both source documents and translations

If the now deprectated **pushTrans** option is used, it is interpreted as **-Dzanata.pushType=both**.

More Information

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=736898.

2.2.4. Project Groups

Versions of multiple projects may have a real-world relationship. For example, the versions of various projects which go into a certain version of Fedora, and need to be translated together. Zanata had no formal support for this type of organization.

Zanata 1.6 introduces Project Groups, which can contain specific versions of various projects.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=800869.

2.2.5. Push individual locale translations with maven

In previous versions of Zanata, only project owners could push translations using maven or the python client. Translators were not able to push translations, making offline translations impractical.

Translators can now push translations for individual locales with the **-Dzanata.locales** option.

Example maven option to push translations for English: **-Dzanata.locales=en,es-ES,en-US**

The specified locales must be a subset of those present in **zanata.xml**, and may be the mapped locale or the local locale.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=736898.

2.3. Web Editor Features

2.3.1. Project-wide Search and Replace: Select all text flows in all documents

In a large project, a global search may return results from many different documents. When using the "Replace all selected" function, there may be too many results to manually select them all.

Zanata 1.6 introduces a "Select All" function. Now you can search through all documents in a project, then select all of the results, and replace all instances.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=817457.

2.3.2. Display page context in window title

Zanata's pages did not have human-readable titles for all pages. Users were unable to distinguish between different open browser tabs, and browsers that show a page title in their history would only show "Zanata" for all pages.

Zanata now shows a specific title for each section of the application, except for the Web Editor, and browser history contains these details.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=813193.

2.3.3. Enhanced Glossary Integration

A Translation Glossary provides standard translations for technical terms. Zanata supports Translation Glossaries. Users requested to be alerted to any applicable Glossary terms when working on a translation unit, and to be able to

copy a translation from the Glossary into the current translation unit.

Users were not alerted when relevant Glossary terms existed, and had to manually copy and paste Glossary terms in the translation editor.

Automatic alerting and copy from Glossary have been implemented. When one or more relevant Glossary terms exist, the color of the Glossary tab changes from blue to red. A "Copy" button next to each Glossary term enables the translator to copy Glossary translations to the current translation unit.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=788294.

2.3.4. Java variable validations in translation editor

Zanata did not detect inconsistencies in Java-style variables between source strings and translated strings. As a consequence, translators could modify Java variables embedded in strings, causing compilation or runtime errors in localised builds of Java software projects.

Zanata 1.6 adds a validator to the web editor to check for Java variables in the source and translated strings.

Java style variables use an index number and may have a format specifier. The validator checks that:

- all index numbers that are present in source are present in target
- all index numbers that are present in target are present in source
- each index number is present the same number of times in the source and target

A validation error is displayed near the string currently being edited if any of the above conditions are not met.

The validator does not yet perform any validation of format specifiers.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=802577.

2.3.5. Offline translation feature via Web Interface

In prior releases of Zanata, translators could download translation work via the Web Interface, but were not able to upload their offline work.

Zanata 1.6 introduces the ability for translators to upload their offline translations in the web interface.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=725319.

2.3.6. Plural Form Support

In translation files, msg_plural entries provide multiple versions of a noun string. These entries allows software to present the correct singular or plural version of the noun to match a number in a message. Zanata did not support plural forms, and silently ignored msg_plural entries in POT/PO files.

Zanata now supports msg_plural entries. Plurals are now accepted/returned via REST clients, accepted/displayed in the editor UI and in TM search results.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=803572.

2.3.7. Project-wide Search and Replace

Translation projects contain a number of distinct files with source material. Applying the same change across a number of files requires opening each file, finding the item that needs to be changed, and applying the change. Zanata's Find and Replace function worked within a file, but not across files.

Zanata now has a "Find and Replace" feature that works across all the files in a project. Users can now apply a change across all the files in the project from a single point in the user interface. Please see the bug report for more details on the state of this feature in the alpha release.

For more information refer to https://bugzilla.redhat.com/show_bug.cgi?id=703806.

2.3.8. Project-wide Search and Replace: Preview of replacement

Zanata includes powerful find and replace functionality that can be applied to individual search results or to a group of search results. It was not possible to see what the effects of a replace operation were before applying it.

Users can now use the preview feature to see the effect of a replace operation before it is carried out.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=822366.

2.3.9. Project-wide search: Search in msgid, msgstr, or both

Searches in Zanata's web editor would search all strings, both source and target strings.

The new search dialog gives users the ability to specify what content they wish to search: source strings only, target strings only, or both.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=817453.

2.3.10. Project-wide Search and Replace: Undo button

Zanata includes a facility for performing search and replace operations, on both single entries and multiple entries. Replace operations could not be programmatically reversed. Users had to manually revert a change, or several changes if multiple search results were replaced at once.

Zanata 1.6 now includes an *undo* for Search and Replace.

- Individual 'Replace' buttons are replaced with 'Undo' buttons when undo is possible (i.e. after a replace has occured).
- An 'undo' link is presented next to the message (bottom right) when multiple translations are replaced.
- Individual 'Undo' buttons can be used after 'Replace All Selected' if the user wants to undo only some of a set of replaced translations.
- Undo is only available until the translation is updated, so the 'Undo' button for a text flow will change back to 'Replace' if another user saves a translation (this is to prevent new translations being overwritten with an older translation).

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=816850.

2.3.11. Search functionality in document list now case-insensitive

Zanata includes a document list filter to allow users to quickly locate specific documents. This filter was case sensitive, which meant users had to remember the exact case used for a document title.

The document list filter now has an option for case-sensitivity. By default the filter is case- *in* sensitive. To perform a case-sensitive filter, check the "case sensitive" check box next to the filter.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=802582.

2.3.12. Short-cut key reference

Zanata has a number of shortcut keys that can be used in the web editor to enhance productivity. There was no single point of reference to discover the available key shortcuts in the web editor. Many key shortcuts can be discovered through tooltips on UI elements, but some key shortcuts cannot be discovered through the UI.

Zanata 1.6 adds a key shortcut summary that shows available key shortcuts for the current view. The summary is invoked by pressing **`Alt+Y`**, and hidden by clicking outside the summary.

The list of shortcuts is not complete. Editor key shortcuts have not yet been added to the summary.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=830000.

Chapter 3. Resolved Issues

3.1. Administrator Issues

3.1.1. Reindex operation takes a long time on production server

Zanata has a number of indexes that are used to locate information on the server. These indexes are periodically updated. The search re-indexing operation always discarded all existing indexes. This caused re-indexing to take a long time, even when only a single class was being reindexed. While the re-indexing operation was taking place, searches did not return good results.

The Search re-index backing beans now take a set of boolean options for each indexable class. These options determine whether existing indexes for the class will be purged, whether re-index will be performed on entities of the class, and whether the index for the class will be optimized. These options are presented in a table on the "Manage Search" page under the Administration menu.

Classes can now be re-indexed separately, allowing for shorter re-index time per operation. Existing indexes can be kept during re-indexing, so searches may be able to return good results during re-index.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=815197.

3.2. Project Owner Issues

3.2.1. Copy Trans ignores some translations when duplicate contents are detected on a single document

When users pushed documents with the same content in different translation units to a new version of a project, copy trans sometimes skipped some of the translation units with duplicate content.

The copyTrans mechanism now accounts for duplicate content scenarios and correctly copies them to the new version.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=825076.

3.2.2. Maven client should print readable error if project ID has spaces

Spaces are not allowed in Zanata project names. When a space was included in a project name, either in **`zanata.xml`** or as a command line argument, the error message did not clearly explain the issue.

When a space is included in a project name now, the error message clearly explains the problem.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=701524.

3.2.3. Server URL in zanata.xml and zanata.ini inconsistent

The server-provided sample **`zanata.xml`** and **`zanata.ini`** files sometimes had mismatched server URLs. This was because the **`zanata.xml`** sample used the server configured URL, and the **`zanata.ini`** used the current navigation URL.

When the two did not match, Zanata clients would not properly associate user configuration from **`zanata.ini`** with the server specified in **`zanata.xml`** leading to error messages about "missing user configuration information".

The server-provided sample **`zanata.ini`** now uses the server configured URL, and the two server-provided sample files now match consistently.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=816453.

3.3. Web Editor Issues

3.3.1. First row missing when moving to a new page of translation

Saving the last translation unit on a page in the web editor moves to the next page. The new page scrolled in such a way that the second row was at the top, and the first row was scrolled off the page.

The web editor's scrolling is modified, and when scrolling to a new page the first row appears at the top of the page.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=803557.

3.3.2. New error message indicator, workspace menu redesign

Zanata provided notifications and error messages in the page header of the editor. The page header took up screen space, which was not available for translation units. Additionally, notifications and error messages would be overwritten by later messages. This message overwriting made it difficult to read a message if it was immediately followed by another.

In Zanata 1.6 the page header uses a single line, leaving more space for translation. Error messages now appear in pop-ups.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=824269.

3.3.3. Save dialog from filtering always saves as approved

When a user checked any of the message filter checkboxes, they were prompted to save their current translation. This translation was always saved as approved.

The Save Dialog that appears when checking a message filter checkbox now contains an additional button "Save as Fuzzy". Users can save their translation as fuzzy by using this button.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=815638.

3.3.4. Search remains "active" after opening new file

The term used in a searches within a document is stored in a history token. When a new document was loaded after a search, the search in the history token was not updated. If the search term was not present in the new document it sometimes made the search box invisible and impossible to clear without manipulating the URL.

The search history token is now cleared with loading a different document.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=803538.

3.3.5. Switching Message Filter before loading completed causes incorrect filter result

If a user changed the message filters while a page was loading, the filter was not applied properly and messages were listed when they should have been filter out.

User input is now is prevented by a modal pop-up during page load, and a user cannot change the filters until page is fully loaded.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=816827.

3.3.6. Translation memory shortcut keys don't work

The code for the translation memory copy shortcut keys Ctrl+Shift+[1-4] checked for characters rather than key codes. The numbers on a keyboard number pad were not recognised, and on some keyboard mappings numbers above 2 were not recognised.

The code was rewritten to recognise keyboard codes, rather than characters. Pressing Ctrl+Shift+[1-4] now copies the translation memory match.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=802191

3.3.7. User security permissions are cached and require logout/login to refresh when changes are made

Zanata performed authorization checks by querying the security entity representing the user. However, when a user is made an owner of the project the user entity is not updated - rather, the project entity is updated. The user entity is only updated with information about project ownership when the user logs in and the user entity is created. When a user was given ownership of a project, they had to log out and log back in again before Zanata would authorize them as the owner of the project.

The authorization check is now made by querying the project entity, rather than the user entity. Users no longer have to log out and log back in again to update their authorizations when they are assigned ownership of a project.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=730199.

3.3.8. Zanata accepts PO files with too few plural forms

Zanata handles msg_plural entries in PO files. A msg_plural entry in a PO file must contain a number of the plural forms that matches the value set in the file for nplural, which is the number of plural forms in the target language. Zanata did not validate the number of plural forms in the PO files against the value of nplural, and incorrectly reported the state of msg_plural entry with too few plural forms.

Zanata 1.6 now correctly identifies and reports when a msg_plural entry has too few plural forms.

For more information see https://bugzilla.redhat.com/show_bug.cgi?id=814006.

Chapter 4. Compiler Output

Topic ID 213

- INFO: Assigned Writer: writer Topic Types: Reference
- ERROR: Topic does not have the fixed url property tag.