MADCAP LINGO 10.2

What's New

CONTENTS

CHAPTER 5

CHAPTER 6

CHAPTER 7

CHAPTER 8

CHAPTER 9

APPENDIX

Introduction

This release of Lingo contains several new features and enhancements.

For more information about each feature discussed in this manual, open the online Help and refer to the "What's New" topic. Links are provided in some feature descriptions, taking you to topics that contain additional information and steps.

This chapter discusses the following:

Summary of Lingo 10 Major New Features

Following are the *major* new features introduced in this version of Lingo.

Feature	What's New?	For More Information
DITA Translation Enhancements	■ Confrefs ■ Dynamic attribute filtering ■ New element types supported	See "DITA Translation Enhancements" on page 12.
Dynamic Filtering	■ Conditions ■ Targets	See "Dynamic Filtering" on page 23.
Flare Integration Enhancements	■ Font mapping ■ Index and glossary sort orders ■ Language skins	See "Flare Integration Enhancements" on page 35.
Flattening	■ Flatten snippets ■ Flatten variables	See "Flattening" on page 44.
Source Control	■ Apache Subversion ■ Git ■ Microsoft Team Foundation Server ■ Perforce	See "Source Control" on page 57.

Summary of Lingo 10 Additional New Features

Following are several *additional* new features and enhancements introduced this version of Lingo.

Feature	What's New?	For More Information
64-Bit Support	▪ Lingo is now 64-bit ▪ Better access to RAM	See "64-bit Support" on page 59.
Command Line	▪ Export a Lingo project ▪ Export a Lingo project in a specific language ▪ Update/sync a Lingo project with the source file	See "Command Line" on page 60.
Dictionaries and Spell Check	▪ Global or project dictionaries ▪ Redesigned Spell Check window pane	See "Dictionaries (Global) and Spell Check Enhancements" on page 61.
File Type Support	▪ Import unrecognized file types as text files ▪ FrameMaker files sent for review ▪ Flare target files for PDF and EPUB	See "File Type Support Enhancements" on page 70.
Help—Live or Local	▪ Choice between live external or local Help system	See "Help—Live or Local" on page 78.
Image Replacement	▪ Batch and multi-language image replacement ▪ Skin image replacement	See "Image Replacement" on page 81.

Feature	What's New?	For More Information
Project Export	■ Update previously exported versions	See "Project Export Enhancements" on page 93.
Project Properties Dialog	■ Basic information ■ Project settings	See "Project Properties Dialog" on page 94.
Reports	■ New QA reports ■ Styling updates	See "Report Enhancements" on page 99.
Smart Quotes	■ Replace quotes based on target language	See "Smart Quotes" on page 106.
TOC View Window Pane	■ Filter files according to TOC from Flare project	See "TOC View Window Pane" on page 107.
Translation Editor	■ Improved conditional text support ■ Cleaner index and concept tags	See "Translation Editor Enhancements" on page 111.

Summary of Lingo 10.1 New Features

Following are the new features introduced in Lingo 10.1.

Feature	What's New?	For More Information
Filter by Multiple Targets	■ Choose more than one target to filter ■ Used when creating project or performing dynamic filtering	See "Filter by Multiple Targets" on page 123.
Show/Hide Spaces in Editors	■ New button in editors ■ Shows or hides spaces between words	See "Show/Hide Spaces in Editors" on page 125.
Microsoft Bing Machine Translation Changes	■ Provide API key ■ Can also provide category	See "Microsoft Bing Machine Translation Changes" on page 127.
Importing Alternate Tag Segments	■ Useful when importing TMX files previously used in third-party tools	See "Importing Alternate Tag Segments" on page 129.

Summary of Lingo 10.2 New Features

Following are the new features introduced in Lingo 10.2.

Feature	What's New?	For More Information
Adobe FrameMaker 2017	■ Support for new version	See "Adobe FrameMaker 2017 Support" on page 131.
Custom Installation	■ Option to install local Help or use server Help only	See "Custom Installation" on page 132.
Find and Replace Enhancements	■ Limit results to the first instance in each file ■ Drop-down with options added to quick find and replace widget	See "Find and Replace Enhancements" on page 134.
Microsoft TFS 2017/2018	■ Support for newer versions	See "Microsoft TFS 2017/2018 Support" on page 138.

CHAPTER 2

DITA Translation Enhancements

Lingo now supports content references and parses them for translation. You can also flatten content references to text and translate them in the Translation Editor. Additionally, many new element types have been added, and you can now filter your project content based on several DITA attributes.

This chapter discusses the following:

Content References

If the conref is flattened, translators can edit the translation to make it more appropriate for the segment where it appears. Flattened conrefs appear as normal text in the Translation Editor, while unflattened conrefs appear as tags. Transltors can move these tags to their correct location in the target segment.

Similarly, you may have multiple versions of content based on the document you are writing. This is often the case if you have content that needs to be translated differently for different audiences.

You can flatten conrefs when creating a new project, or you can change an existing project's flattening settings using the Project Properties dialog. Settings are updated dynamically, so you do not need to update the project to apply the settings.

☆ EXAMPLE

You use content references throughout your DITA project. You know that your existing translation memory (TM) already includes some of the segments in your project, but the segments in your TM do not use content references; instead, the text is written as part of the segment. Because of this, you decide to flatten the content references in your entire project. This way, when you send the project to your translator, she will get the most accurate matches from your existing TM. This will also let her account for context in the segment when translating the content references.
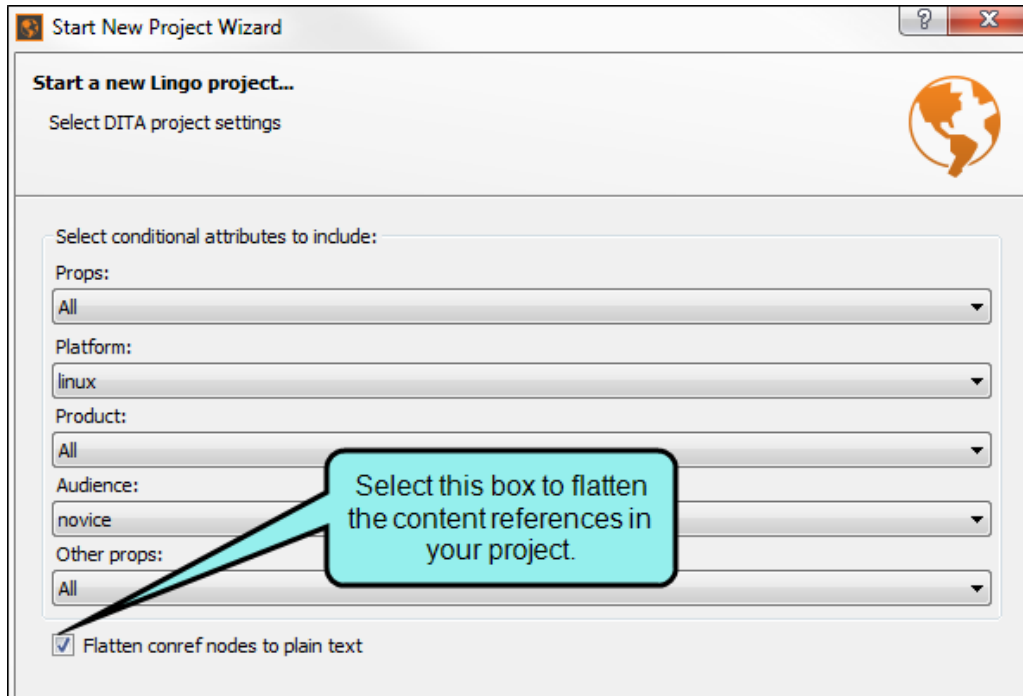
NOTE: The DITA file associated with a conref must be part of your ditamap, otherwise Lingo will not be able to locate the file when importing the DITA project.

NOTE: As a best practice, you should always translate conrefs in the associated DITA file, even if you have flattened them and translated them in the segment. Lingo will use the translation from these files for instances of the conref that are not included in the segment (e.g., conrefs that are removed from the Translation Editor).

NOTE: You can flatten both inline and block conrefs. Flattening block conrefs may add additional tags to the segment in order to preserve the formatting found in the block of content.

Dynamic Attribute Filtering

You can filter DITA files to include one or more conditional attributes. The resulting Lingo project will include all of the translatable files that are marked with the selected attributes (while excluding those that are not selected). Additionally, because Lingo supports block-level and inline attributes, content within a segment that is marked with an excluded attribute will also be excluded from a filtered Lingo project.

If you use conditional attributes to filter your Lingo project, the filtered content and files will be used throughout your project when running reports, creating bundles, and performing other tasks. For example, if you pull a report, you will only see data from the filtered content. If you make a bundle, you will only be able to select from files that are included in the project.

You can filter a project when creating a new project, or you can make changes to an existing project's filtering settings using the Project Properties dialog. Settings are updated dynamically, so you do not need to update the project to apply the settings.
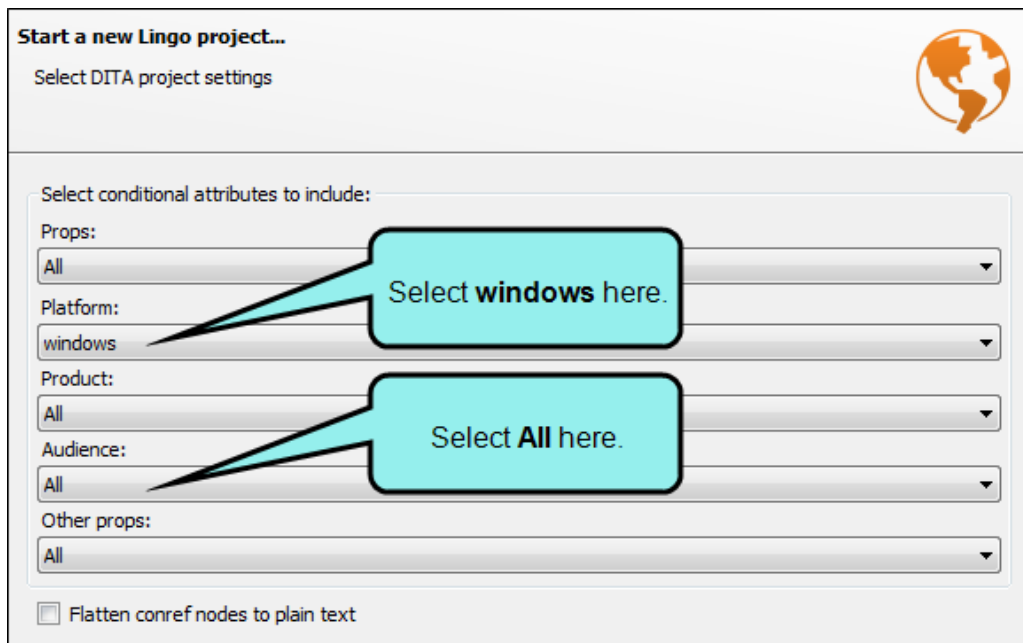
☆ EXAMPLE

Your DITA files include information for Mac, Windows, and Linux operating systems. In your DITA files, you used the platform attribute to mark the content applicable to each of these operating systems. Additionally, you used the audience attribute to mark content for expert and novice users.

```xml
1    <?xml version="1.0" encoding="UTF-8"?>
2    <!DOCTYPE map PUBLIC "-//OASIS//DTD DITA Map//EN" "map.dtd">
3    <map>
4      <title>DITA Topic Map</title>
       <topicref href="windowsexpert.dita" audience="expert" platform="windows">
         <topicref href="topic1.dita"/>
         <topicref href="topic2.dita"/>
       </topicref>
9      <topicref href="windowsnovice.dita" audience="novice" platform="windows">
10       <topicref href="topic5.dita"/>
11       <topicref href="topic6.dita"/>
12     </topicref>
13     <topicref href="linuxexpert.dita" audience="expert" platform="linux">
14       <topicref href="topic3.dita"/>
15       <topicref href="topic7.dita"/>
16     </topicref>
17     <topicref href="linuxnovice.dita" audience="novice" platform="linux"/>
18     <topicref href="macexpert.dita" audience="expert" platform="mac">
19       <topicref href="topic4.dita"/>
20     </topicref>
21     <topicref href="macnovice.dita" audience="novice" platform="mac"/>
22     <topicref href="expert1.dita" audience="expert"/>
23     <topicref href="expert2.dita" audience="expert"/>
24     <topicref href="novice1.dita" audience="novice"/>
25     <topicref href="novice2.dita" audience="novice"/>
26   </map>
```

Your ditamap looks like this.
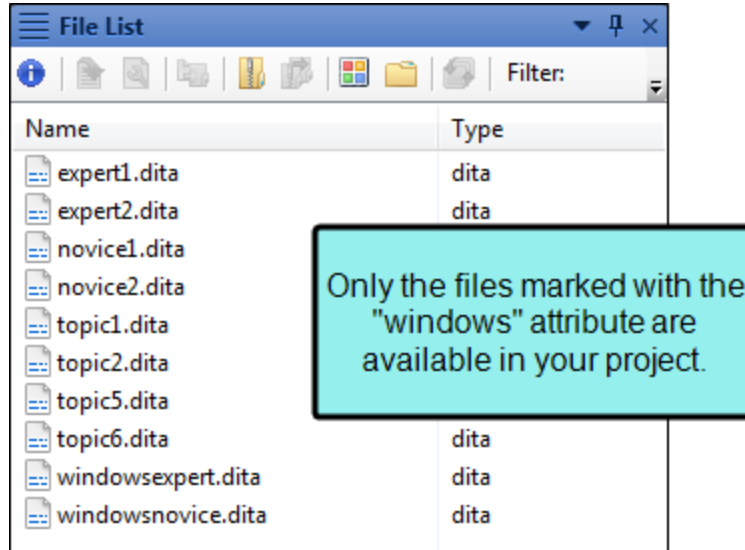
☆ When creating your Lingo project, you have the option to filter your content by either the platform or audience attributes. You could also filter the content using both attributes.

You need to translate all of the content that is applicable to all of your users who use Windows. You select the **windows** platform attribute to include this attribute type (excluding Mac and Linux) and leave the audience attribute set to **All** (including both expert and novice users).

⭐ Your project looks like this in Lingo:

File List

Name	Type
expert1.dita	dita
expert2.dita	dita
novice1.dita	
novice2.dita	
topic1.dita	
topic2.dita	
topic5.dita	
topic6.dita	dita
windowsexpert.dita	dita
windowsnovice.dita	dita

Only the files marked with the "windows" attribute are available in your project.

☆ You then use Lingo to create a bundle and send the Windows files to your translator.

Bundle Settings

◉ Lingo Project Bundle

○ XLIFF Files Bundle

☐ Include Project Translation Memories

☐ Include Project Termbases

Bundle Files

Project Files Filters:

All File Types ▾ All Translation Statuses ▾ All Difference St ▾

Project Files

Name	Path

Output Files

Because only the files with the "windows" attribute were filtered into the project, only these files can be added to the bundle. This makes it easy to send them to the translator.

Name	Path	#
windowsnovice....		2
windowsexpert....		2
topic6.dita		2
topic5.dita		2
topic2.dita		2
topic1.dita		2
novice2.dita		2
novice1.dita		2
expert2.dita		2
expert1.dita		2

☆ You also need to translate content for expert Linux users, but this content goes to a different translator. You open the Project Properties dialog and adjust the filter settings so the platform attribute is set to **linux** (excluding Mac and Windows) and the audience attribute is set to **expert** (excluding novice users).
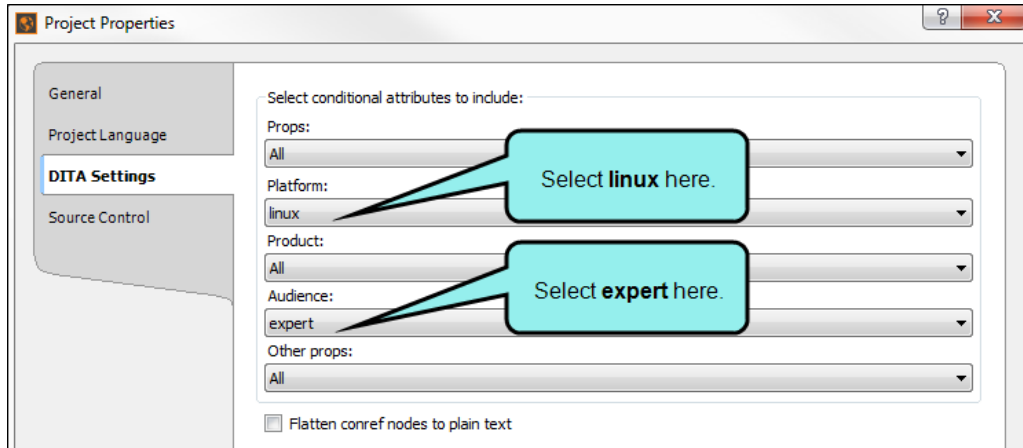
Your project dynamically updates, and now it looks like this in Lingo:
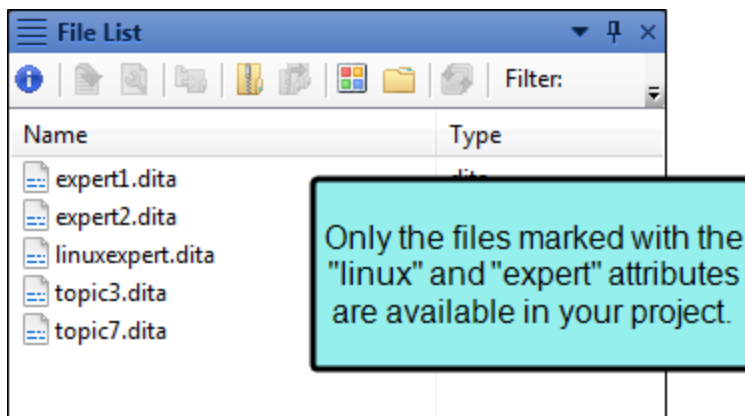
Because your project now displays only the expert-level Linux files, you can create a bundle with these files and send them to the other translator.

📝 NOTE: If you update the filter settings for an existing project, the translation status of changed segments is set to Untranslated. You also have the option to completely clear any existing translations for these segments.

New Element Types Supported for Translation

Lingo now parses several new DITA element types for translation. The following elements are now translatable:

- <alt>
- <author>
- <codeblock>
- <desc>
- <glossaryacronym>
- <glossarysurfaceform>
- <glossdef>
- <glossterm>
- <indexterm>
- <mainbooktitle>
- <ph conref>
- <publisher>
- <refsyn>
- <related-links conref>
- <stepresult>
- <xref>

> **NOTE:** For more detailed information about DITA, please refer to:
>
> www.oasis-open.org/committees/dita/.

Dynamic Filtering

You can now filter a source Flare project by target or conditions. Filtering by target allows you to create a Lingo project that includes files in a specific Flare target, while filtering by condition removes files and content marked with specific conditions. Using filtering options narrows down large Flare projects or more easily breaks down large projects into smaller chunks based on conditions or targets.

You can filter a project when creating a new project, or you can make changes to an existing project's filter settings using the Project Properties dialog. Settings are updated dynamically, so you do not need to update the project to apply the settings.

This chapter discusses the following:

Filtering Out Conditions

You can filter Flare projects to exclude one or more conditions. You can also select conditions based on the entire condition set, or you can pick the specific conditions in the set that you want to filter by. The resulting Lingo project will include all of the translatable files that are not marked with the selected conditions. Additionally, because Lingo supports block-level, inline, glossary, snippet, and CSS conditions, content within a file or segment that is marked with an excluded condition tag will also be excluded from a filtered Lingo project.

If you use condition tags to filter your Lingo project, the filtered content and files will be used throughout your Lingo project when running reports, creating bundles, viewing glossary terms, and performing other tasks. For example, if you pull a report, you will only see data from the filtered content. If you make a bundle, you will only be able to select from files that are included in the project.

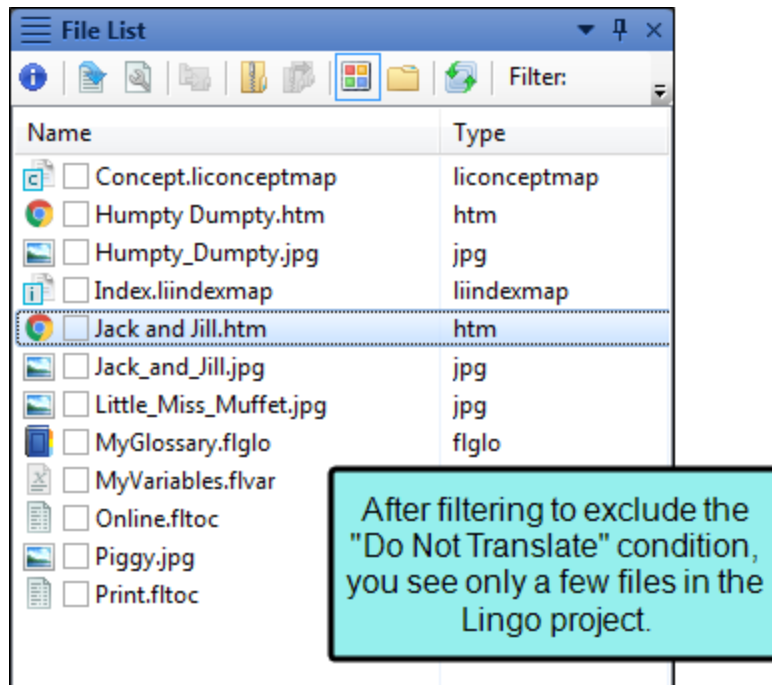⭐ EXAMPLE

You are an author with a very large Flare project, but you only need to send specific files to your translator. Rather than exporting your entire project to Lingo and then searching for the individual files that you need to export for translation, you instead mark the files and folders you do not need to translate with a "Do Not Translate" condition tag as you create each item in Flare.
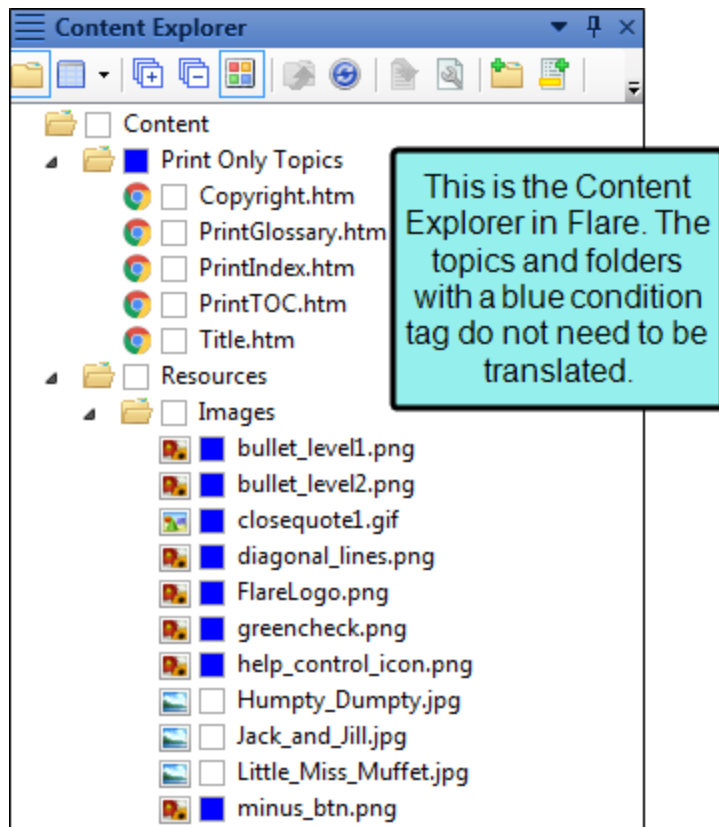
☆ In Lingo, you then filter the project to exclude the condition tag, creating a new Lingo project that only includes the files you need to send to your translator.

Exclude the "Do Not Translate" condition tag.
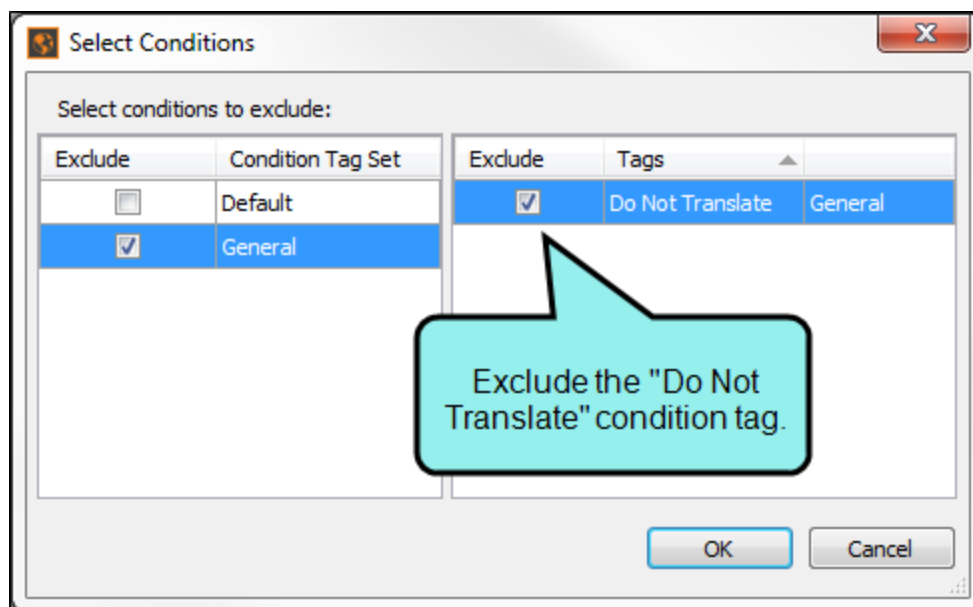
☆ When you open the project in Lingo, it looks like this:

Because you applied the "Do Not Translate" condition throughout your project, you see only a handful of topics and images. You also see a few project files, such as your glossary, variables file, and TOCs.

⭐ EXAMPLE

The content author filtered her Flare project to exclude a "Do Not Translate" condition tag. Of the nine lines in the original topic (including the heading), four were marked with a "Do Not Translate" condition tag.
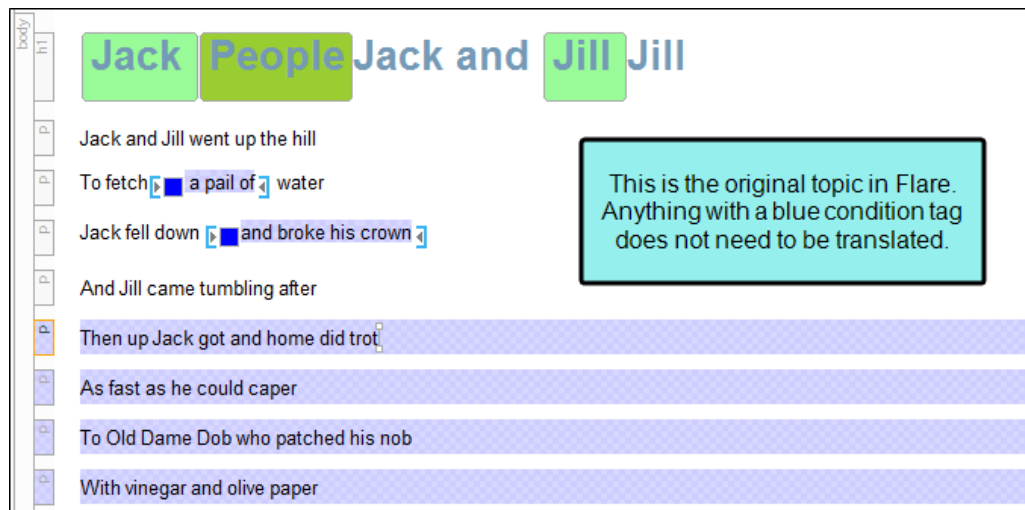
When you view the topic in the Translation Editor, only five segments appear in Lingo. This is because four lines were excluded when the project was created. Additionally, you see blue condition tag indicators in some segments where individual words were conditioned out and excluded from a segment.
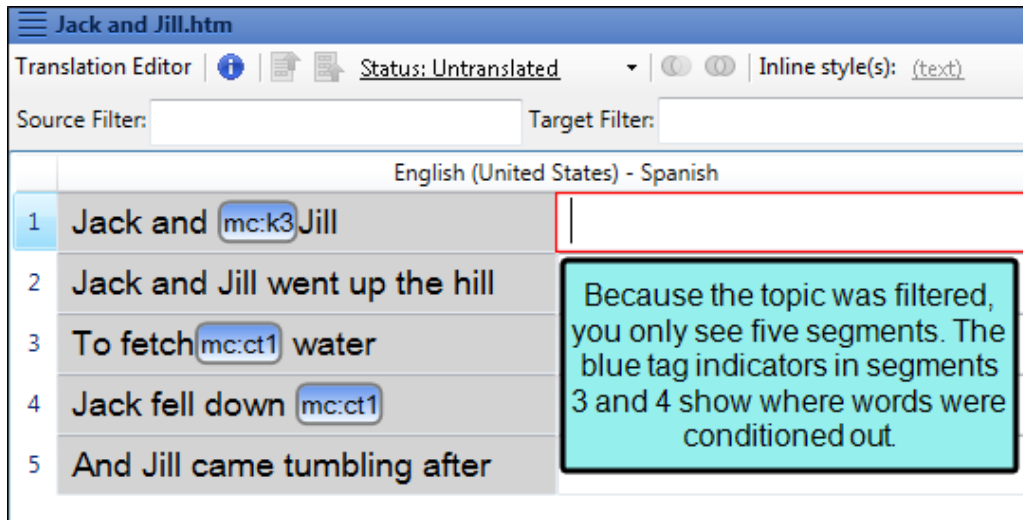
NOTE: Conditions that "unbind" rather than "exclude" content will not be excluded when you filter out a condition in Lingo. If you notice that content has not been removed but you think that it should be, check to be sure that the condition is not marked as an "unbind" condition in Flare.

Filtering by Target

You can filter your Flare projects by one or more targets. The resulting project will include all of the translatable files that are part of the selected target(s), such as topics, images, skins, and variables.

A good reason to use target filters is if you need to maintain different translations for different targets. Additionally, you could filter to a specific target and use variable flattening if you need use different definitions of a variable in different targets.

> ☆ **EXAMPLE**
>
> You are preparing Lingo projects for your translators. You have two translators: a senior translator who translates your PDF manuals and online Help; and an assistant translator who translates shorter documents, like brochures.
>
> When you import your Flare project to Lingo, you decide to filter by target. Rather than creating a big Lingo project and then trying to remember which files go in the brochures and which go in the online Help, it's faster for you to use the targets you already created in Flare to quickly filter the project.
>
> First, you need to send files to your assistant translator, who is working on a brochure about lists. To do this, you filter by the Brochure target.

☆ Lingo automatically adds the correct files to the project, and you can easily bundle the files and send them to the assistant translator.

Click the bundle icon to bundle these files and send them to the translator.

You only see the files that are part of the Brochure target.
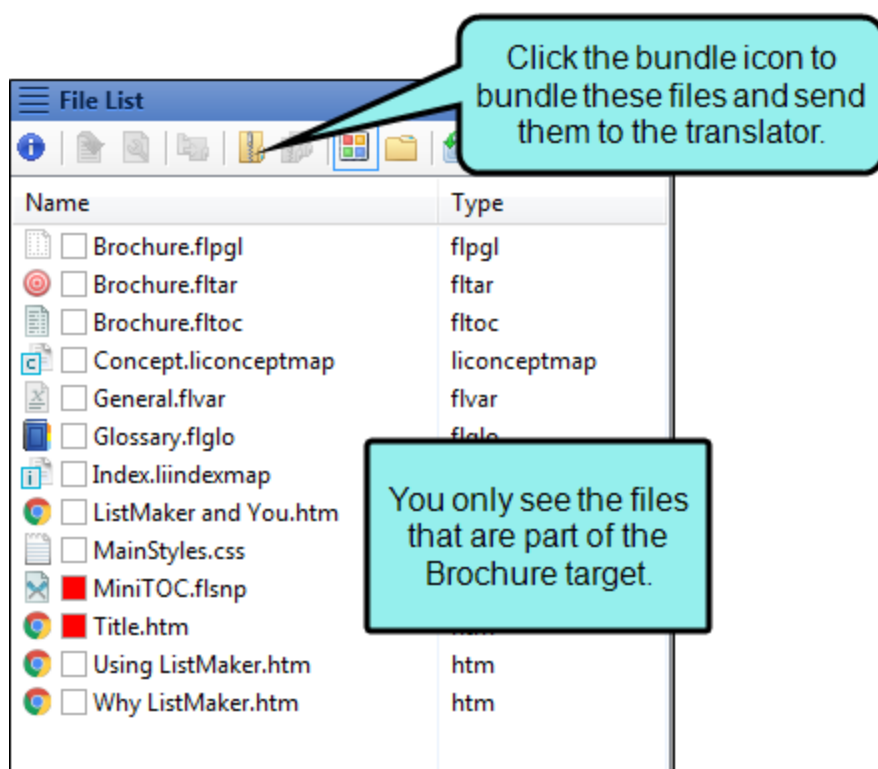
☆ Additionally, since the variable definitions used in the project are determined by the target filter, your translator will see the correct variables when translating the project.

🗒 NOTE: If you filter by target, target-specific variables can be translated in the target file from the File List. Defau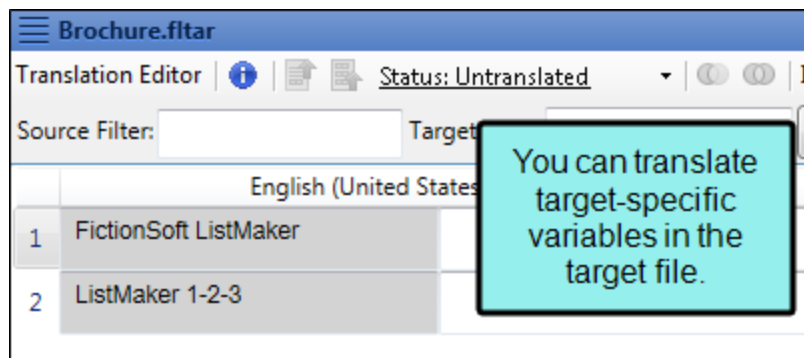lt variable definitions are translated in the variable set file (they are not considered target-specific, even if selected for a target).

All other variables (and variables in projects not filtered by target) can be translated in the variables file.

As a best practice, you should translate variables in the variable set or target file, even if you have flattened them and translated them in the segment. Lingo will use the translation from these files for instances of the variable that are not included in the segment (e.g., variable tags that are removed from the Translation Editor).

Settings

You can filter conditions and targets when creating a new project, or you can update an existing project's filtering settings on the Flare Settings tab of the Project Properties dialog (**File>Project Properties**). The settings are the same in both locations.

- **Filter by target** Select this option, then select the target whose files you want to include in the Lingo project. Any files that are not part of that target will not be shown in the File List. If you want to replace variable definitions for a specific target, you can select variables from the **Select variables to flatten** grid.

- **Filter out condition** Select this option, then click **Select conditions**. In the dialog that opens, select the condition tag set(s) you want to exclude from the Lingo project from the grid on the left side of the dialog. Then select or deselect individual condition tags in each set from the grid on the right side of the dialog.

> **NOTE:** You can also filter DITA projects using conditional attributes.

> **NOTE:** If you update the filter settings for an existing project, the translation status of changed segments is set to Untranslated. You also have the option to completely clear any existing translations for these segments.

Flare Integration Enhancements

Enhancements have been made to improve the way Lingo projects integrate with MadCap Flare. These features include the ability to create font maps, sort index and glossary terms, and create language skins.

This chapter discusses the following:

Font Mapping

If you are translating a project into multiple languages, you may need to use different fonts for each language. Using fonts appropriate for the language's character set ensures that the characters display correctly when viewed in the output. You can use Lingo's font mapping feature to locate the fonts already used in the source Flare project and map them to fonts that are supported by each target language.

To do this, open the Manage font Maps dialog (**Resources>Manage Font Mapping**) and create a font map, then edit it. In the dialog that opens, select the files you want to scan. Lingo will search these files and list the fonts currently being used. Then you can select the font you want to map to each original font. You can do this for each language in your project.

When the translated project is exported, the Flare project will use the mapped fonts for each language.

⭐ **EXAMPLE**

You are translating a Flare project that was originally written in English. You need to translate it into French and Arabic. You know that each of these documents will use different fonts in the final output, so you use Lingo to create a font map.

First you scan your project files for fonts. The files are currently using Arial, Calibri, and Cambria. This is fine for French, which does not need to display a special character set, and therefore does not need a special font. However, you need to use a different font for Arabic in order to support the language's character set. You switch the language to Arabic and map each of the fonts in your project to B Fantezy, which is an Arabic font.

When the author exports the Arabic project back to Flare, the output uses B Fantezy and all of the characters are supported.

This HTML5 output uses the mapped Arabic font.

مقالة تغيير الخط

خط معين في المقالة

خط معين في ملف التخصيص

NOTE: Lingo scans the following file types for fonts:

- Master pages (.flmsp)
- Page layouts (.flpgl)
- Snippets (.flsnp)
- Stylesheets and table stylesheets (.css)
- Topics (.htm)

NOTE: Some fonts (like Arial) include characters used in other languages. However, there is no guarantee that these fonts will work well in all programs or in all browsers. For best results, you should always map fonts to an appropriate character set

Index and Glossary Sort Orders

There may be times that you want to create a custom sort order for your Flare index or glossary. You can do this by adding a "Sort As" term to the index or glossary term. Custom sort orders are especially important when translating a Flare project because terms in each translated language may have different spellings or pronunciations that are not automatically picked up by Flare. Because of this, you may need to specify how the index or glossary should sort a term, so the term is identified correctly when the Lingo project is exported back to Flare.

After you translate the index or glossary file, open the Project Properties dialog and select the **Sort Order** tab. From here, you can see all of the index or glossary terms in your project, as well as the custom terms set for the source project, if applicable. If any of the translated terms need to be sorted using a different letter, you can add an entry in the **Sort As** column. When the author exports the translated project to Flare, the index or glossary will use the new terms to alphabetize the entries.

☆ **EXAMPLE**

You are translating a project into Japanese, which uses three different character types. You are translating mostly in kanji, but you want to use katakana in your index and glossary because these characters are phonetic.

Because of this, you use Lingo's sort as feature to add katakana "Sort As" terms for each index and glossary word in the Lingo project.

Select file:

Index.liindexmap

Source Entry	Source Sort As	Entry - Japanese	Sort As - Japanese
Boss		上司	じょうし
Company		会社	かいしゃ

These are the translated entries. They use the kanji alphabet.

These are the Sort As entries. They use the katakana alphabet.

☆ Later, when the project is exported back to Flare, the index and glossary are sorted using the katakana alphabet instead of the kanji alphabet.

キーワード

This is how the index looks when the author generates PDF output in Flare. Notice that the keywords--although still displayed in kanji--are sorted using the katakana alphabet.
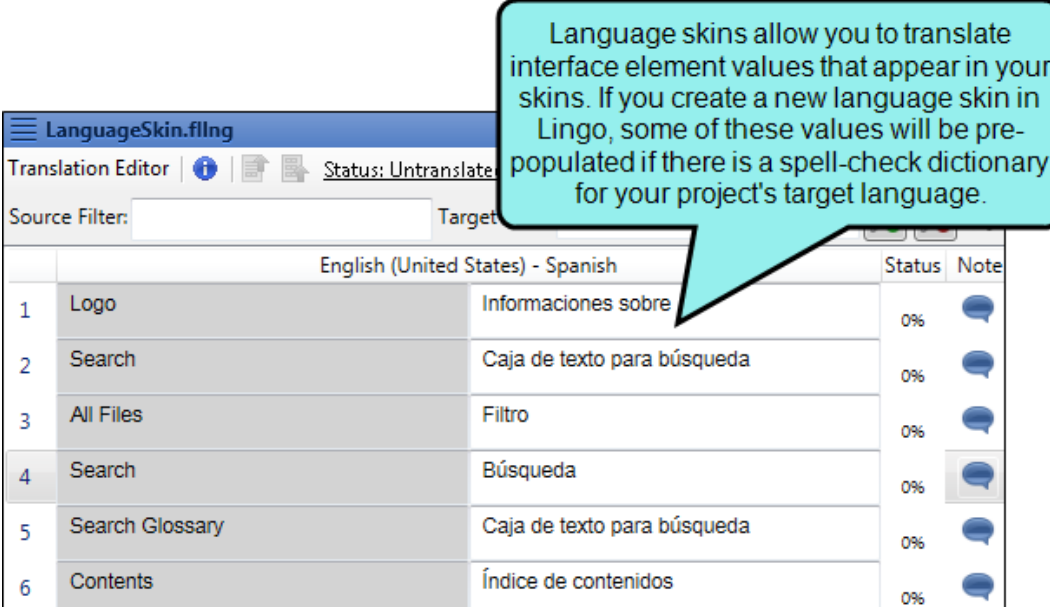
か

会社 2

し

上司 2

📝 NOTE: If your project has been filtered by a target or condition tag, the glossary file only displays terms that have not been excluded.

Language Skins

When you create your Lingo project, you have the option to add a language skin to your project. This generates a language skin file (LanguageSkin.fllng) containing several terms used in different Flare skins (e.g., "Table of Contents," "Glossary," "Home"). Translators can open the language skin file to translate these terms. This is especially useful if there are custom terms in the Flare skin and they need to be translated to match the custom interface.

When you export the project back to Flare, the language skin can be seen by opening the Project Organizer, then selecting the **Advanced** folder. Additional changes can be made to the language skin in Flare, if necessary. When the output is generated, the translations in the language skin replace Flare's default translations for interface elements.

> NOTE: If your project's target language has an available spell check dictionary, some translations will be pre-populated when you create your language skin. The available translations vary depending on the language you are using. Languages with an available spell check dictionary are displayed in bold when you select a language during project creation.

Language skins allow you to translate interface element values that appear in your skins. If you create a new language skin in Lingo, some of these values will be pre-populated if there is a spell-check dictionary for your project's target language.

NOTE: When translating language skin segments, the interface element value is added to the segment as a segment note for your reference.
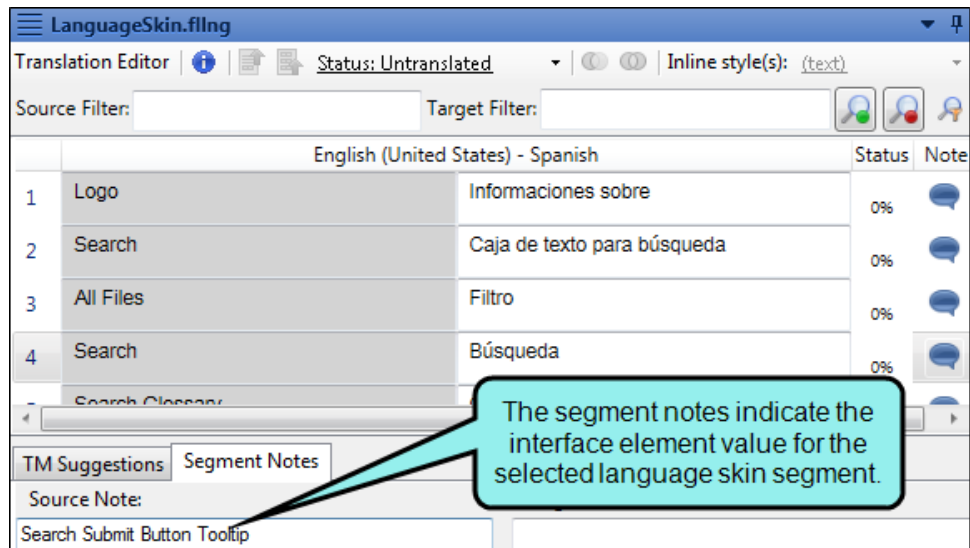
NOTE: You can only add language skins to a project if there are not language skins already in the source Flare project. If the source Flare project already includes one or more language skins, they will be added to the Lingo project and can be translated.

Flattening

You can now replace Flare variables and snippets with text in the Translation Editor. This is referred to as "flattening."

If the content is flattened, translators can edit the translation to make it more appropriate for the segment where it appears.

Similarly, you may have multiple versions of content based on the document you are writing. This is often the case if you have content that needs to be translated differently for different audiences.

You can flatten snippets and variables when creating a new project, or you can change an existing project's flattening settings using the Project Properties dialog. Settings are updated dynamically, so you do not need to update the project to apply the settings.

This chapter discusses the following:

Flattening Snippets

You can flatten sub-segment snippets (i.e., snippets that are part of a segment, as opposed to their own tag). This means that snippet content will be included as text in the segment in Lingo. Translating a snippet as part of the segment allows translators to account for context, rather than translating each snippet individually. This also makes sure that you have the most accurate translations in the translation memory.

Flattened snippets appear as normal text in the Translation Editor, while unflattened snippets appear as tags. Translators can move these tags to their correct location in the target segment.
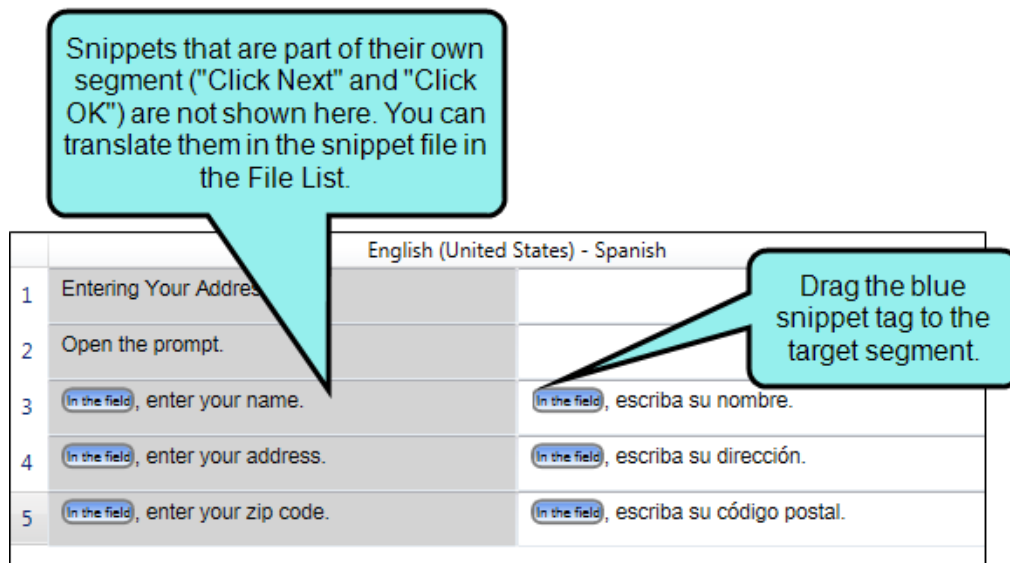
☆ EXAMPLE

There are several short snippets in your Lingo project. Sometimes they appear as part of a segment. Other snippets appear as their own segments.
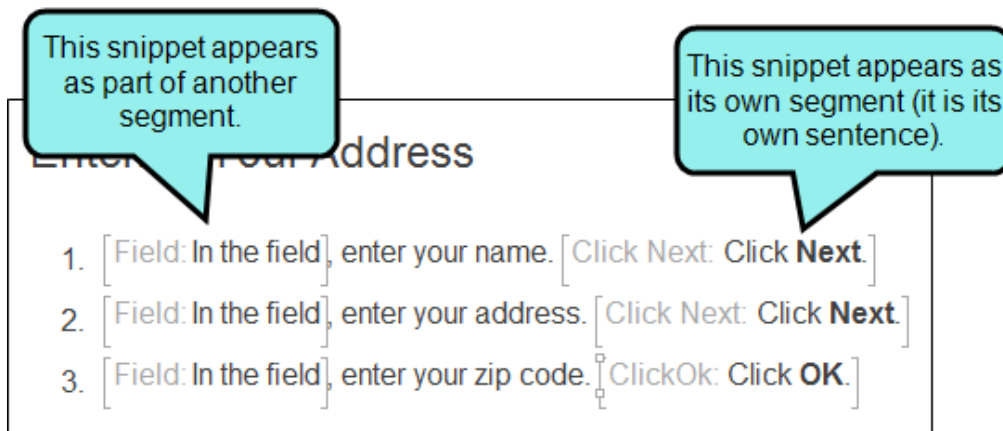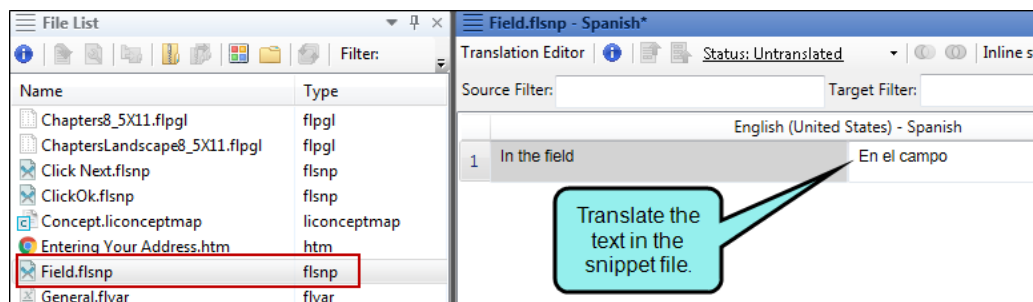
☆ If you do not flatten the snippets...
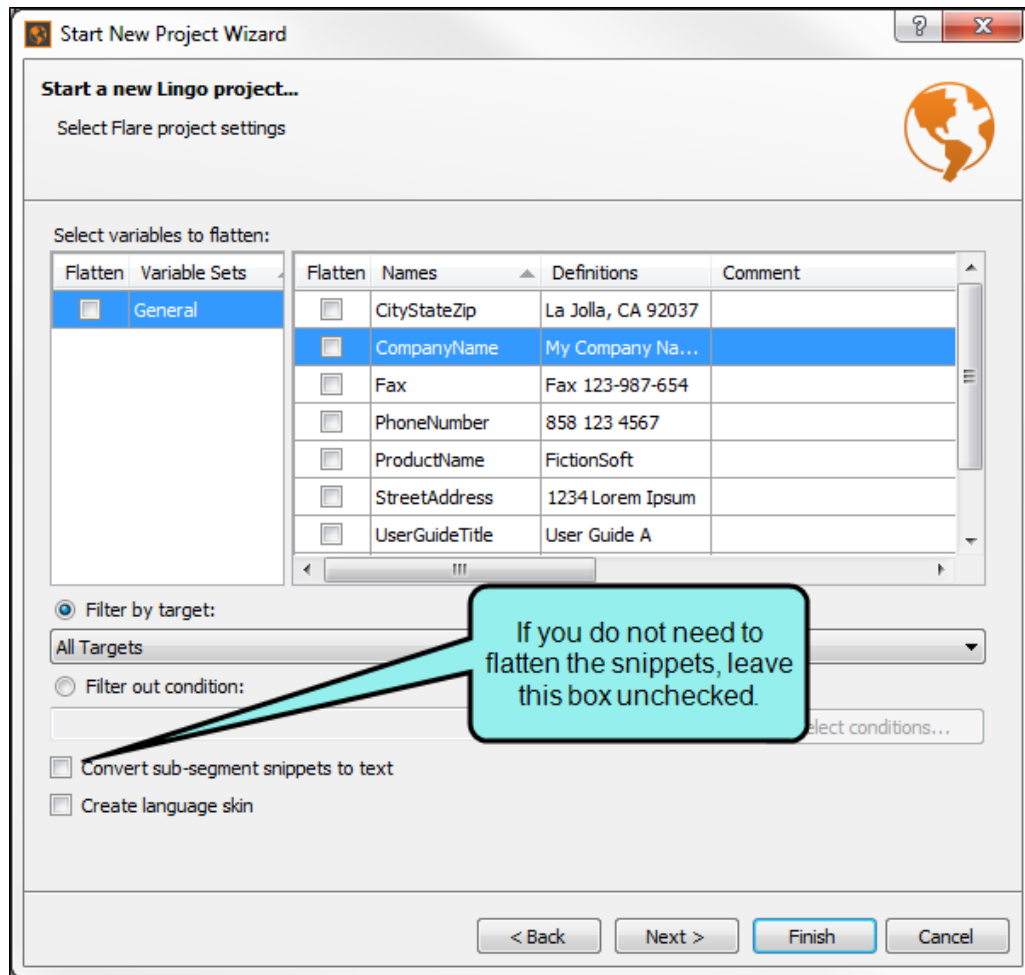
☆ ...the content in Lingo looks as follows. Because the snippets are unflattened, you see tags where the snippets appear in the segments. You can drag each tag into the target segment. Snippets that are part of their own segment do not appear as tags in the Translation Editor.

Snippets that are part of their own segment ("Click Next" and "Click OK") are not shown here. You can translate them in the snippet file in the File List.

Drag the blue snippet tag to the target segment.

You can translate all of the snippets in the snippet files, which are found in the File List.

Translate the text in the snippet file.

☆ If you do flatten your snippets...
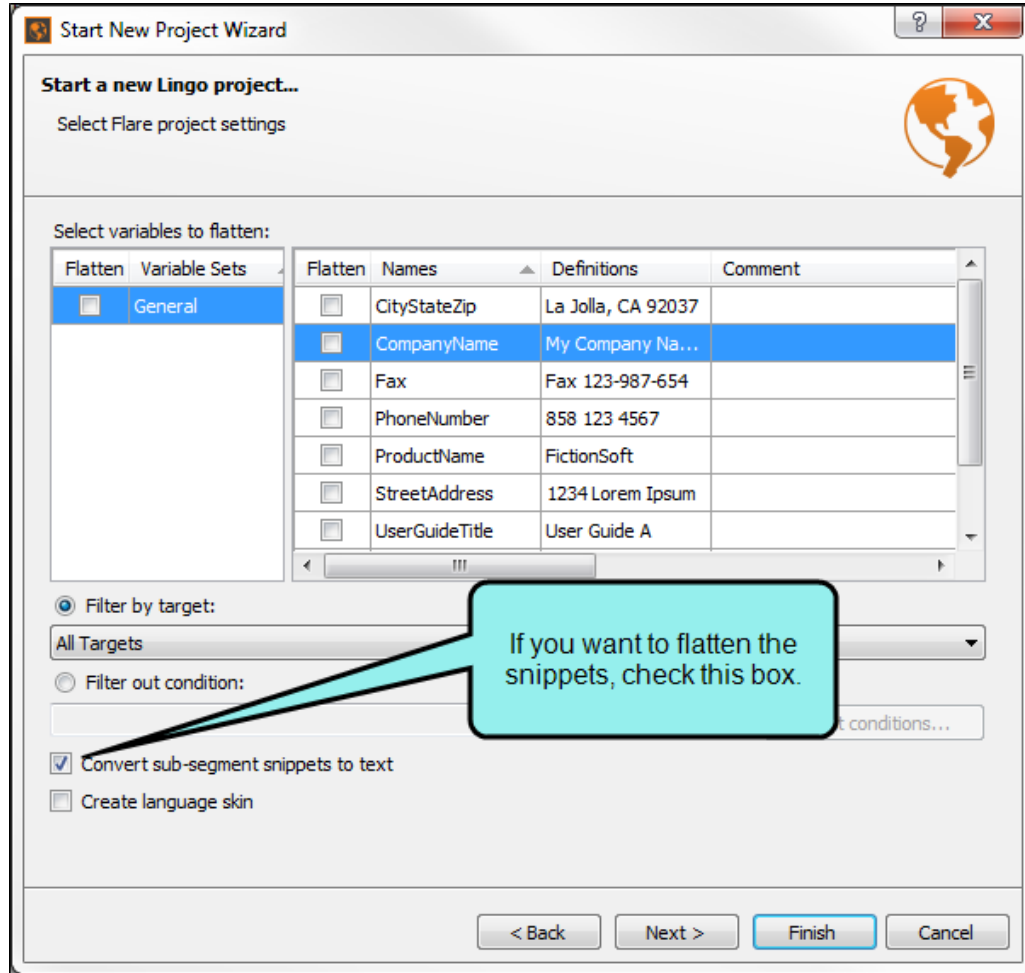
☆ ...the content in Lingo looks as follows. Content from each snippet appears in the Translation Editor as text.
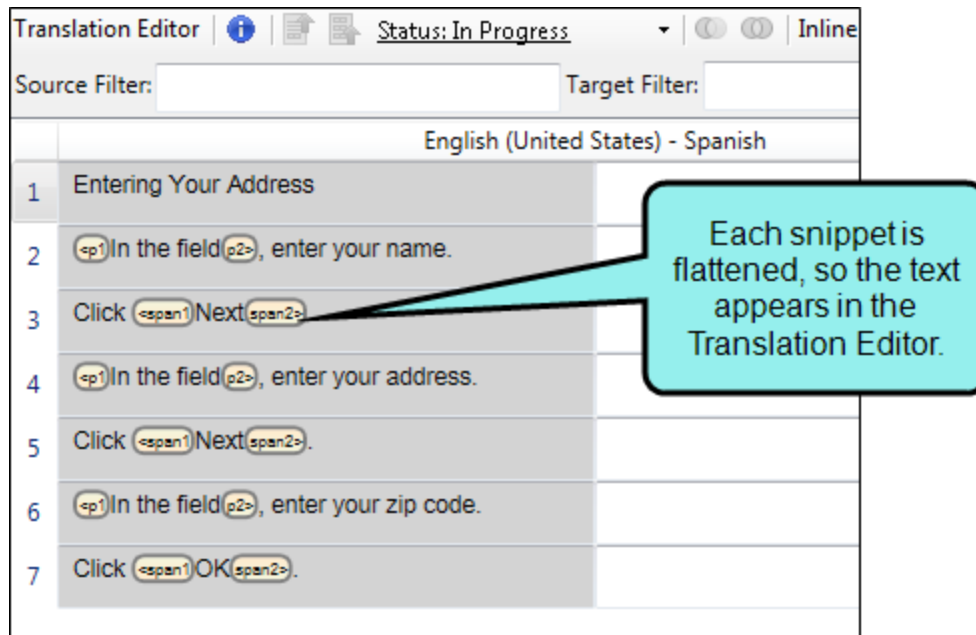
NOTE: You can flatten both inline and block snippets. Flattening block snippets may add additional tags to the segment in order to preserve the formatting found in the block of content.

NOTE: As a best practice, you should always translate snippet content in the snippet files, even if you have flattened them and translated them in the segment. Lingo will use the translation from these files if it finds instances of the snippet that are not included in the segment (e.g., snippet tags that are removed from the Translation Editor).

Flattening Variables

Replacing variables with their definitions is referred to as "flattening." You can flatten variables from a specific variable set, and you can select certain variables in a set. This gives the translator greater flexibility when translating, because some variables (like a company name) may never need to be translated, but others may always need to be translated.

Flattened variables appear as normal text in the Translation Editor, while unflattened variables appear as tags. Translators can move these tags to their correct location in the target segment.

Additionally, you can use variable flattening in conjunction with target filtering if you need use different definitions of a variable in different targets.

☆ EXAMPLE

You are a Flare author writing a short manual for a company that has two internal teams: Red and Blue. One of the variables in your project is a target-specific "Team" variable with two definitions: "Red" and "Blue."

You need your manual translated into Spanish. When creating your Lingo project, you filter the Flare project by target so you only see the appropriate topics and variables for each team.

The first document you produce is the one for the Blue team. To do this, you filter by the "Blue" target when creating the Lingo project. You do not flatten the variables for this project.

☆ The variable definition for "Blue" that was specified in your Flare project is included in the Lingo project. When finished, you bundle the project and send it to your translator.

Because you did not flatten the variables in the project, your translator will translate the variable in the target file (because it is target-specific).
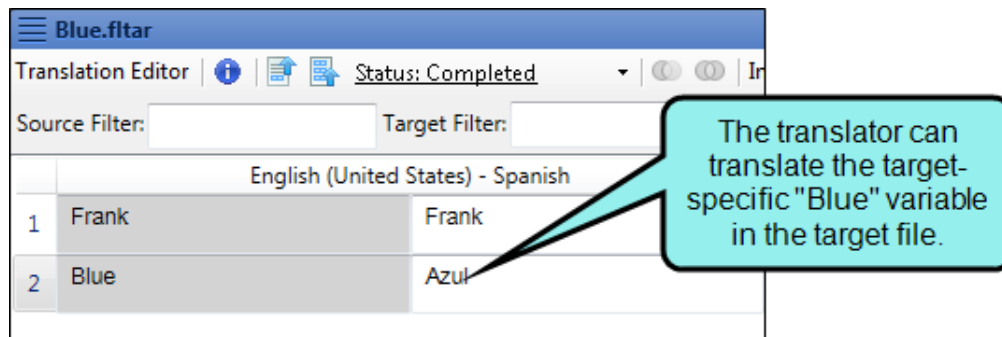
☆ When translating segments, he simply moves the "Blue" tag in the source segment to the target segment. He does not need to translate the variable each time he encounters it in a topic.


When you create the Lingo project for the Red team, you decide to flatten the "Red" variable. Unlike "Blue," which does not take on a different gender depending on the words it modifies, "Red" might be masculine or feminine. You know that you may need to change the variable slightly to use correct gender agreement.

☆ To do this, you update the project's settings on the Flare Settings tab of the Project Properties dialog (**File > Project Properties**). You choose to filter by the Red target. You also select the "Team" variable (which has the "Red" definition in this target) from the **Select variables to flatten area** at the top of the tab.
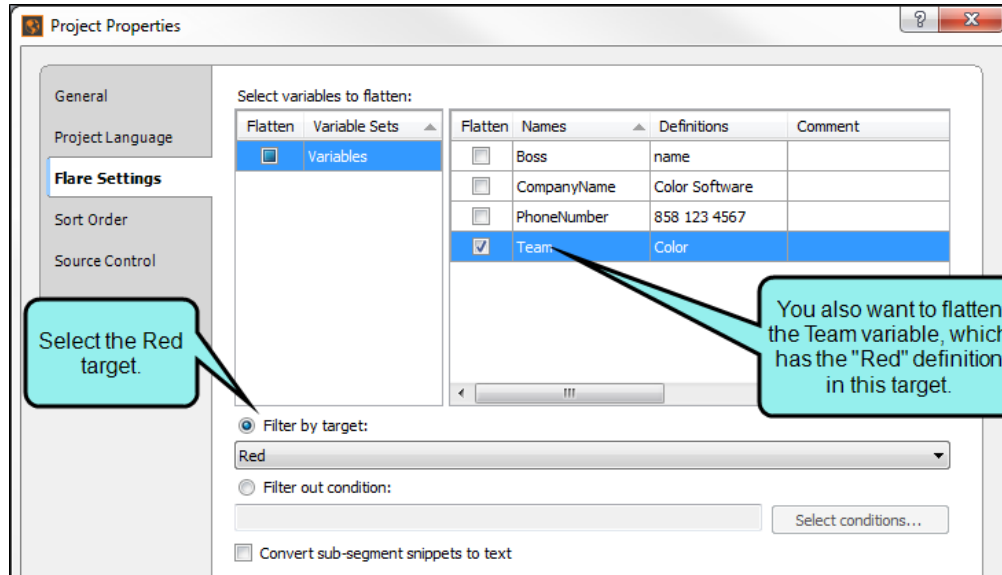
☆ When finished, you bundle the project and send it to your translator.

By flattening the variable, your translator can translate it directly in the segment and make changes based on the context of the sentence.
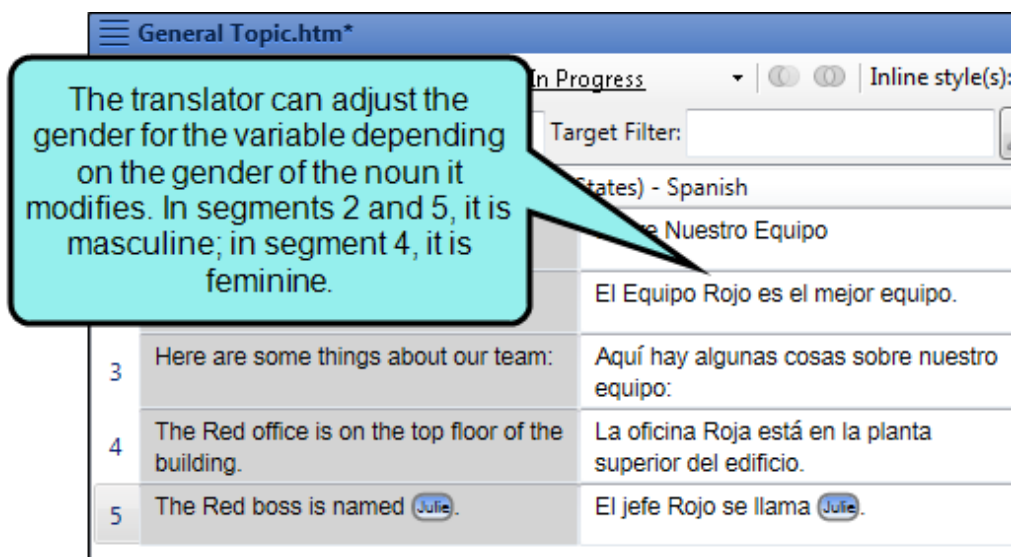
The variable also appears in the target file, because Lingo will use this translation if the variable is used elsewhere in the project.

🗒 NOTE: If you filter by target, target-specific variables can be translated in the target file from the File List. Default variable definitions are translated in the variable set file (they are not considered target-specific, even if selected for a target).

All other variables (and variables in projects not filtered by target) can be translated in the variables file.

As a best practice, you should translate variables in the variable set or target file, even if you have flattened them and translated them in the segment. Lingo will use the translation from these files for instances of the variable that are not included in the segment (e.g., variable tags that are removed from the Translation Editor).

Settings

You can flatten snippets and variables when creating a new project, or you (or the translator) can update an existing project's flattening settings on the **Flare Settings** tab of the Project Properties dialog (**File>Project Properties**). The settings are the same in both locations.

- **Flatten a Variable** Select the variable set(s) you want to flatten from the grid on the left side of the dialog. Then select or deselect individual variables from the grid on the right side of the dialog. If you want to replace variable definitions for a specific target, you can optionally select a target from the **Filter to target** drop-down.

- **Flatten a Snippet** Select **Convert sub-segment snippets to text** at the bottom of the dialog. This will convert all inline snippets to text.

> **NOTE:** If you update the flattening settings for an existing project, the translation status of changed segments is set to Untranslated. You also have the option to completely clear any existing translations for these segments.

> **NOTE:** You can also flatten DITA content references.

Source Control

Lingo provides integrated source control support for Subversion, Git, Team Foundation Server, and Perforce.

Source control is a process where you store a copy of your project on a server, allowing you to "check in" and "check out" files and keep your local changes in sync with the server version fo the project, and with changes from other writers. Source control is even useful for single authors because it provides a means for maintaining a constant backup of all project files.

For more information about Lingo's integrated source control support, please refer to the online Help.

> ⊘ IMPORTANT: If you are using source control, do not include your local termbase (.litdb3) and translation memory (.db) files in your project folder. These files do not function correctly when used with source control, and can cause conflicts. Instead, you should use a server termbase or translation memory. This will allow other users to have access to the terms in the database while avoiding possible conficts.

Additional New Features

In addition to the features already discussed, some additional new features are being introduced in this version.

This chapter discusses the following:

64-bit Support

Lingo is now a 64-bit application. This is beneficial because 64-bit applications provide better access to your computer's memory (RAM). This is especially useful if you have a large project or often work with multiple files at once, because Lingo will run faster and more efficiently.

> **NOTE:** If you have any mappings to the Lingo application (e.g., shortcuts), you will need to update them so they point to the Program Files folder. Previous versions of Lingo were native 32-bit applications and were installed in the Program Files (x86) folder.
>
> For example, the new location of the Lingo application is **C:\Program Files\MadCap Software\MadCap Lingo 10\Lingo.app\Lingo.exe**.

> **NOTE:** If you want to import Doc-To-Help projects to a 64-bit version of Lingo, you will need to download Microsoft Access Database Engine 2010 x64 edition. If you do not already have this add-in, Lingo will prompt you to install it from the Microsoft Download Center.

Command Line

Lingo now supports the command line for managing your projects. Depending on the command(s) used, you can accomplish any of the following.

- Export a Lingo project
- Export a Lingo project in a specific language
- Update/sync a Lingo project with the source file

The best way to use the command line feature is to create a batch file with the necessary commands in it. Then you can use a scheduling tool (such as the Task Scheduler utility in Windows) to run the batch file automatically whenever you want.

> NOTE: Arguments in the command line are case sensitive.

Dictionaries (Global) and Spell Check Enhancements

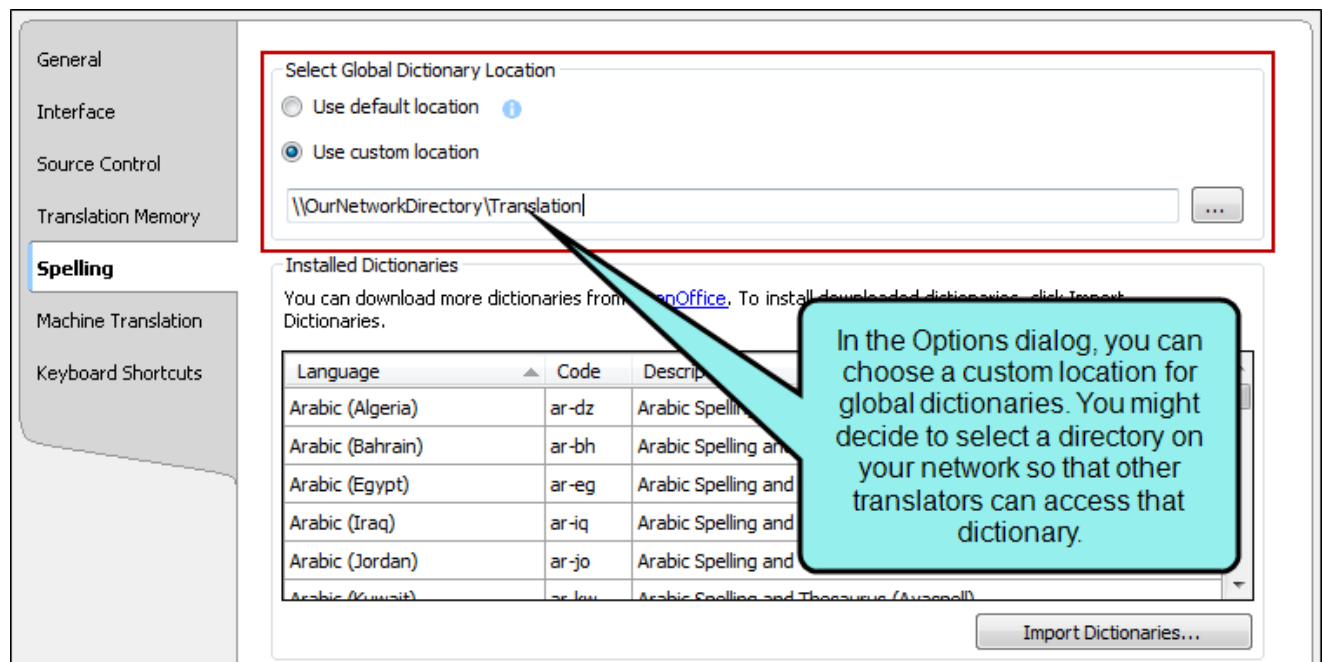Some enhancements have been made to dictionaries and the spell checking feature they support.

Dictionaries—Global or Project

In previous versions, Lingo supported only project-level custom dictionaries for spell checking purposes. By "project-level," we mean that the dictionaries you created were tied to specific projects. These types of dictionaries are still supported. But now you also have the option of creating global dictionaries.

A global dictionary is a file that contains words and can be used for spell checking by any Lingo project you open on your computer. This is a convenient way to ensure that all of your projects are using the same spellings for terms, rather than having each project rely on a different dictionary.

By default, a new global dictionary is stored in your AppData folder, like all of your built-in dictionaries.

However, you can use the Options dialog to choose another location for your global dictionary. For example, you might be working with a team of translators and want to make sure you are all using the same spellings for certain terms. By choosing a network directory, your entire team can use the same global dictionary.

In the Options dialog, you can choose a custom location for global dictionaries. You might decide to select a directory on your network so that other translators can access that dictionary.

You can even use a global dictionary as well a project dictionary. If this is the case, the Spell Check window pane combines and displays suggested spellings from both dictionaries, as well as from the built-in dictionary for that language.
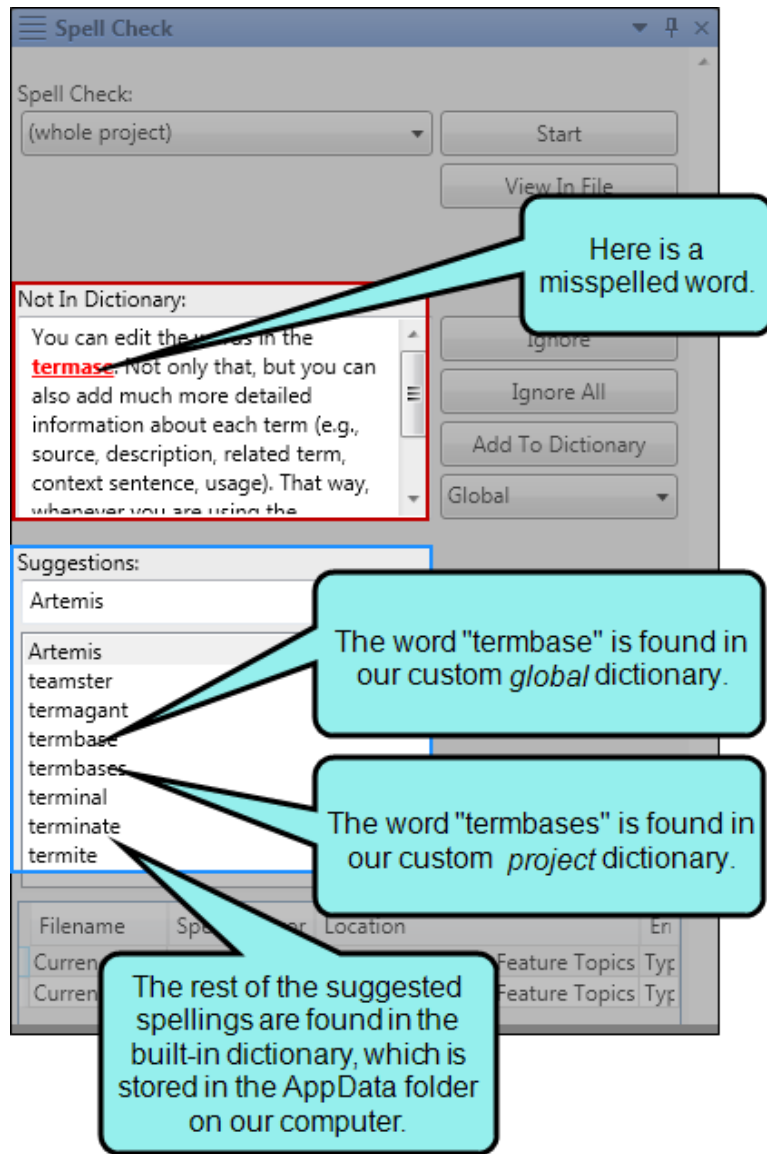
Spell Check Changes

The Spell Check window pane has been redesigned, and several new options have been added to it. This includes the following:

- Look in more locations (e.g., content document, documents in the same folder, whole project).

- Add spellings to global or project dictionaries.

- Open the Options dialog to set global dictionary settings, as well as to tell Lingo to ignore certain types of words or circumstances during spell check.

The process has also changed somewhat. Previously, you used the Spell Check window pane to move from file to file to see each questionable spelling. Starting with this version, the questionable spellings are listed in a grid at the bottom of the window pane after you start the spell check process. Click on a row to deal with a specific misspelling, then select the following row to move to the next one, and so on.

This is desirable because you can perform all of your spell check work in one place, without having to open a bunch of different files. However, you always have the option to open a particular file where a questionable spelling occurs.
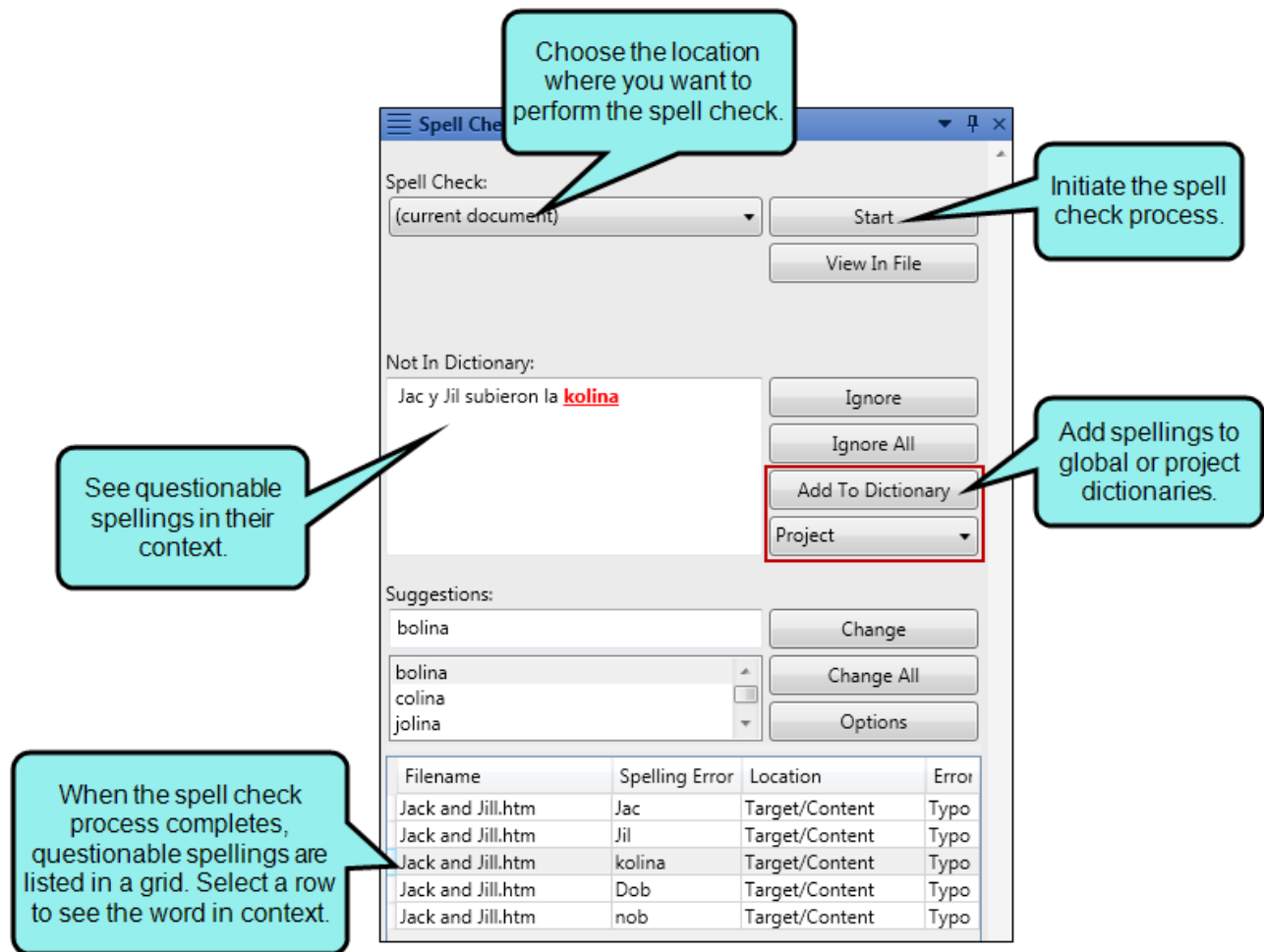
Also, if you press F7, the Spell Check window pane opens and automatically displays the first questionable spelling it finds in the current file.

Following are the main sections in the redesigned Spell Check window pane. Many of the fields and options in these sections are new or changed.

SPELL CHECK

In the first drop-down,select the location where you want to perform the spell check (if you have a Lingo project open):

- **current document** Lingo checks the spelling in the active file only.

- **documents in the same folder** Lingo checks the spelling in all files located in the same folder as the current one.

- **whole project** Lingo checks the spelling in every file in the project.

You can use the buttons to the right to initiate or stop the spell check process, as well as open the file where a questionable spelling occurs:

- **Start/Cancel** Display all occurrences of questionable spellings in a grid at the bottom of the window pane. Each time you select a row in the grid, the "Not In Dictionary" and "Suggestions" fields are populated with the results.

  If the process is in progress, the label changes to "Cancel," which you can click to stop the process.

- **View In File** Open the file where the current questionable spelling occurs, and highlight it.

When you select a questionable spelling in the grid below, the word and its surrounding context is shown here.

If you want to display the entire file where the questionable spelling is found, click this button.

After clicking this row, the fields above were populated to show us the questionable word and suggested spellings.

Spell Check

Spell Check:

(current document) ▾   Start

View In File

Not In Dictionary:

Jac y Jil subieron la **kolina**

Ignore

Ignore All

Add To Dictionary

Project ▾

Suggestions:

bolina

Change

bolina
colina
jolina

Change All

Options

Filename	Spelling Error	Location	Error
Jack and Jill.htm	Jac	Target/Content	Typo
Jack and Jill.htm	Jil	Target/Content	Typo
Jack and Jill.htm	kolina	Target/Content	Typo
Jack and Jill.htm	Dob	Target/Content	Typo
Jack and Jill.htm	nob	Target/Content	Typo

NOT IN DICTIONARY

This field displays each questionable spelling in its context with red, underlined font.
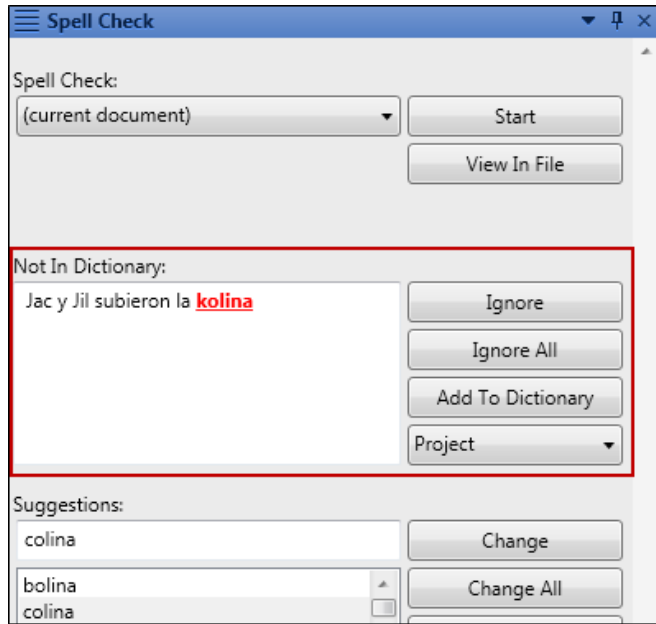
You can use the buttons and drop-down field to the right in order to take action on the questionable spelling.

- **Ignore** Lingo ignores the spelling of the word for the current session.

- **Ignore All** Lingo ignores the spelling of the word and adds the spelling of this word to the Ignored Words Editor (which can be opened from the Tools ribbon). This means that if other occurrences of the word are found in any other files in future sessions, they will not be flagged as questionable spellings. This is similar to adding words to a dictionary. The difference is that this feature simply ignores questionable spellings of certain words, whereas a dictionary also lists spellings as suggestions.

- **Add To Dictionary** Lingo adds the word to the dictionary so that it is not flagged as a questionable spelling in the future. The word will also be displayed as a suggestion for future questionable spellings that are very similar. The kind of dictionary (Global or Project) to which it is added is determined by the selection in the next field.

- **Global/Project** If you select Global, spellings of words are added to a dictionary that can be used by any projects pointing to it. If you select Project, spellings of words are added to a dictionary that can be used only by the current project.

SUGGESTIONS

This field displays alternative suggestions for each questionable spelling.

You can use the buttons to the right in order to replace the current spelling with a suggestion, or to set other options.

- **Change** Lingo changes only the current instance of the misspelled word with the alternative word selected in the Suggestions field.

- **Change All** Lingo changes all instances of the misspelled word with the alternative word selected in the Suggestions field.

- **Options** This opens the Spelling tab of the Options dialog.

  If using a global dictionary, you can use fields at the top of the dialog to choose a location for it.

  You can also use fields at the bottom to ignore any of the following when spell checking files:

  - Ignore repeated words
  - Ignore words with numbers
  - Ignore uppercase words
  - Ignore URLs
  - Ignore mixed case words
  - Ignore email addresses

> ⊙ **WARNING:** When you use a spell check suggestion to change the spelling in a file, the edit is made on disk, not in the editor. Therefore, you cannot undo the action if you change your mind. Instead, you will need to open the affected file and change the text manually.

File Type Support Enhancements

Improvements have been made to Lingo's file type support. Flare target files are now supported, and Framemaker files—previously supported only for translation—are now also supported for review. Additionally, you can now import any unrecognized file type, including custom file types, by using a text file filter.

Import Unrecognized File Types as Text Files

Lingo now allows you to import any file type as a text file by using a text file filter and regular expressions. This is helpful if you work with files that use uncommon or custom file extensions.

To import any file type, create a new Lingo project and add your file to Lingo. If you add a file type that Lingo does not recognize, you will see a warning image next to the file filter drop-down.

To resolve this error, select an existing text file filter or create a new one using regular expressions. Adding a text file filter will treat the file as a text file and allow you to import and translate the file. If you do not select a filter, the file will be discarded when you create the project.

You can also add unrecognized file types as part of a folder. If you do this, Lingo will notify you to apply a text filter to the appropriate files within the folder. If you do not apply a filter to the unrecognized files, they will be discarded from the folder upon project creation.

> ⊘ **IMPORTANT:** Creating file type filters is recommended for advanced Lingo users. This feature requires an understanding of advanced topics such as regular expressions, XML, and segmentation rules.
>
> Additionally, it is recommended that you are familiar with the file(s) you are importing prior to adding a filter. Some file type filters require you to select the specific content that you want to import (e.g., sheet labels, headers and footers, comments).

⭐ EXAMPLE

You need to translate an ODF file. However, Lingo does not recognize this file type. When you add the file to your project, you see a yellow warning icon.
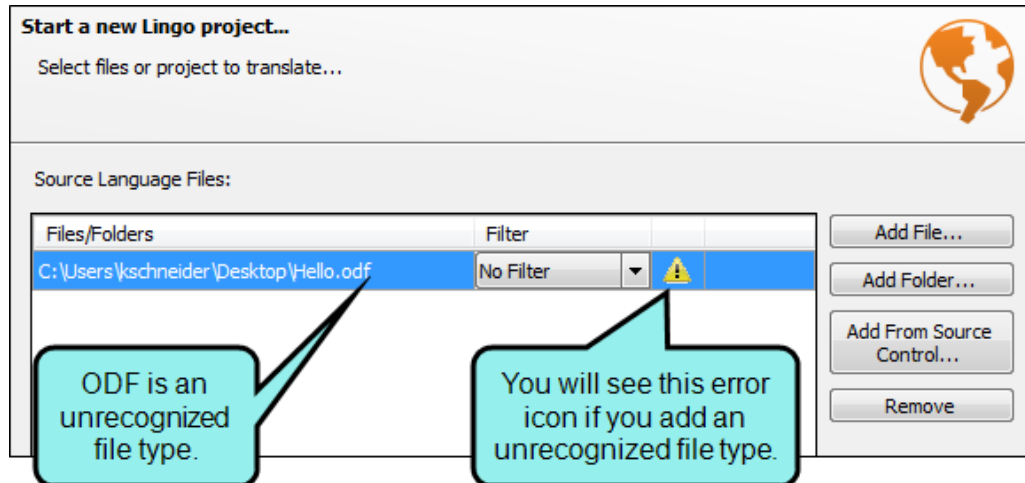
To resolve this error, select the **Filter** drop-down and select **Create New**.
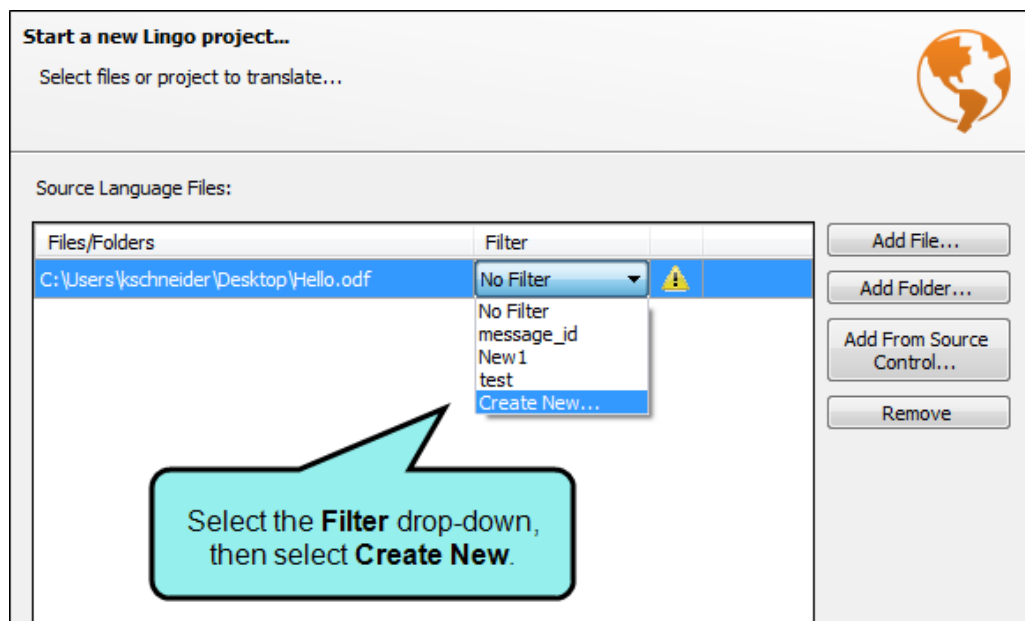
☆ The File Filters dialog opens, and you can create a new text file filter using regular expressions.
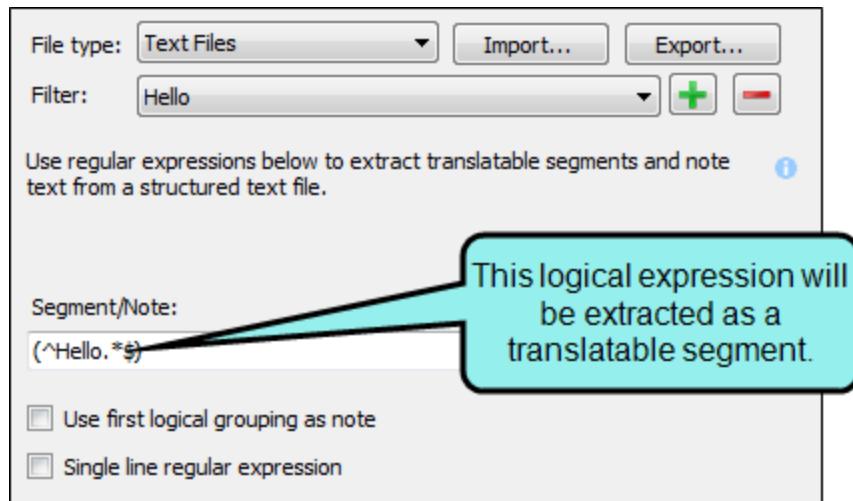
When you select the new text file filter from the drop-down, the error is resolved.
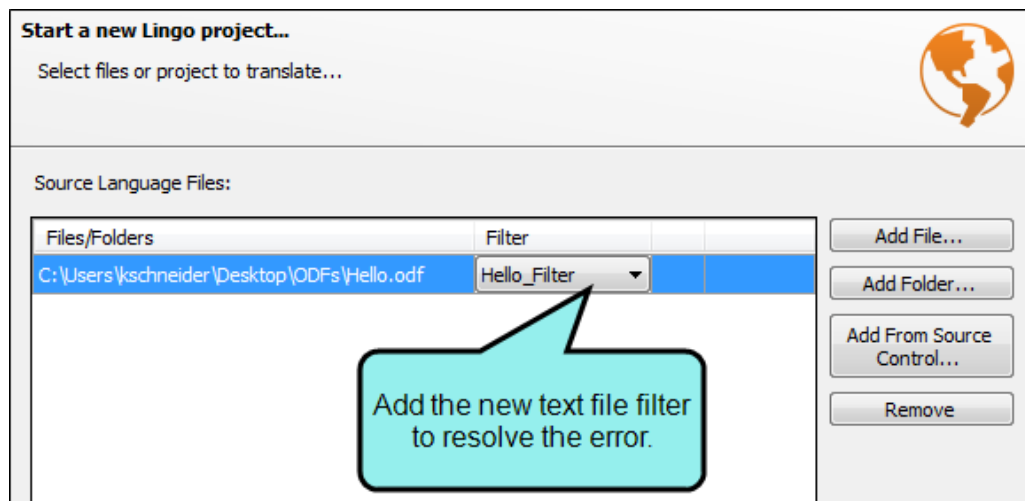
☆ The file is imported for translation.
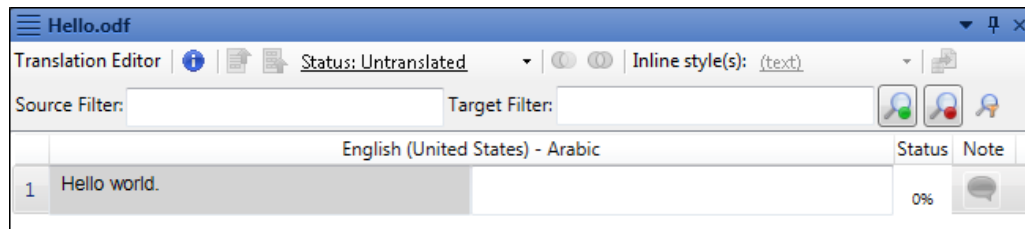
FrameMaker Files Sent for Review

In previous versions of Lingo, FrameMaker files were supported for translation, but you could not send translated files for review. In Lingo 10, you can now send translated FrameMaker files for review using Lingo's internal review process.

When working with FrameMaker files in Lingo, you can save these files as a review package (**File>Review>Save Review Package**). In the dialog that opens, select the files you want to send, then click **Finish**. The files will be saved as a Lingo review package (.lirev).

Flare Target Files

You can now translate Flare target files (.fltar) for PDF and EPUB targets. This allows you to translate the target's metadata fields, such as the document title or description. This metadata is important when distributing content, because it helps publishers correctly categorize information, and it helps users find the document.

You can translate the following metadata:

- PDF
  - Keywords
  - Subject
  - Title
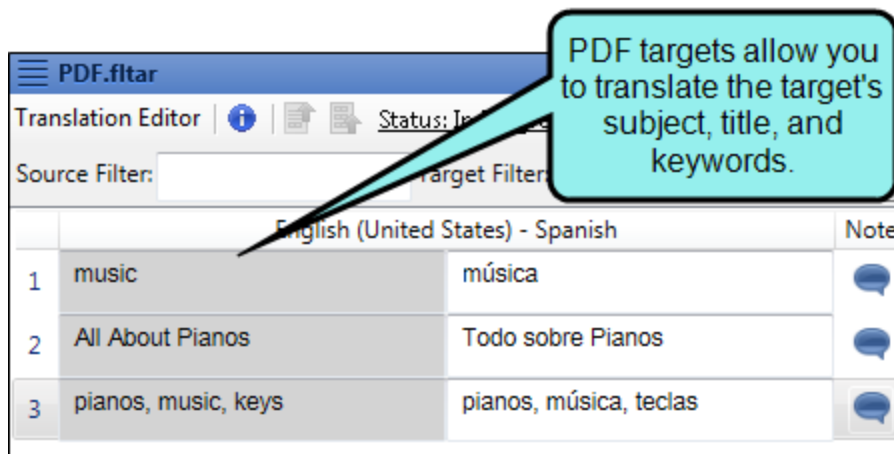
- EPUB
  - Description
  - Genre
  - Subject
  - Title
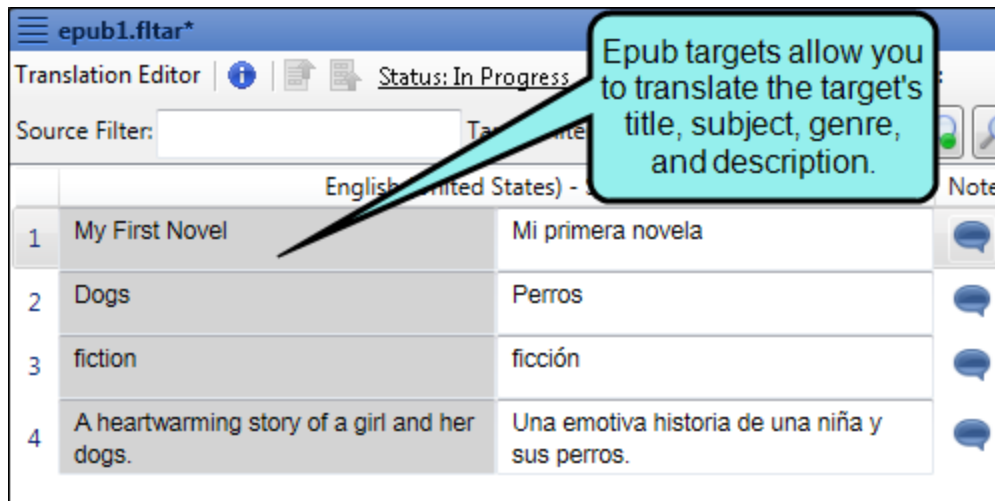
NOTE: When you translate metadata in a PDF or EPUB target file, the metadata type is added to the segment as a segment note.
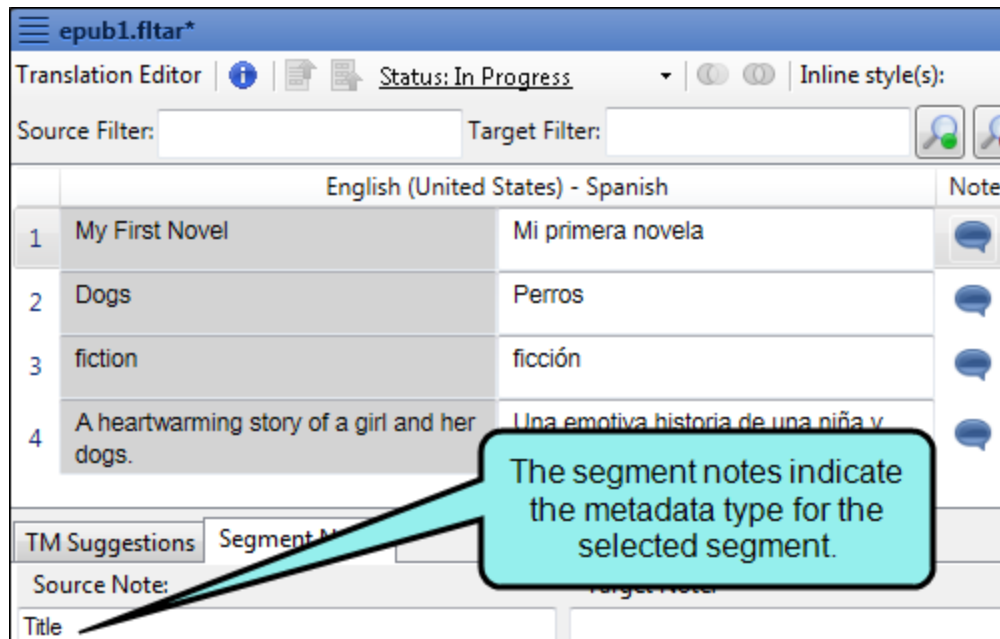
Help—Live or Local

Lingo's online and dynamic Help system has changed, pointing to files hosted externally. This allows for live updates to the Help. Also, there is a new local Help option that lets you view Help located on your computer instead.

Live External Help System

By default, Lingo's dynamic Help is now hosted online, rather than inside the application. This allows topics to be updated on an as-needed basis with the most current information. In previous versions of Lingo, the dynamic Help was only updated when there was a new product release.

To open the dynamic Help for the active element in the interface, select **Help>Dynamic Help**.
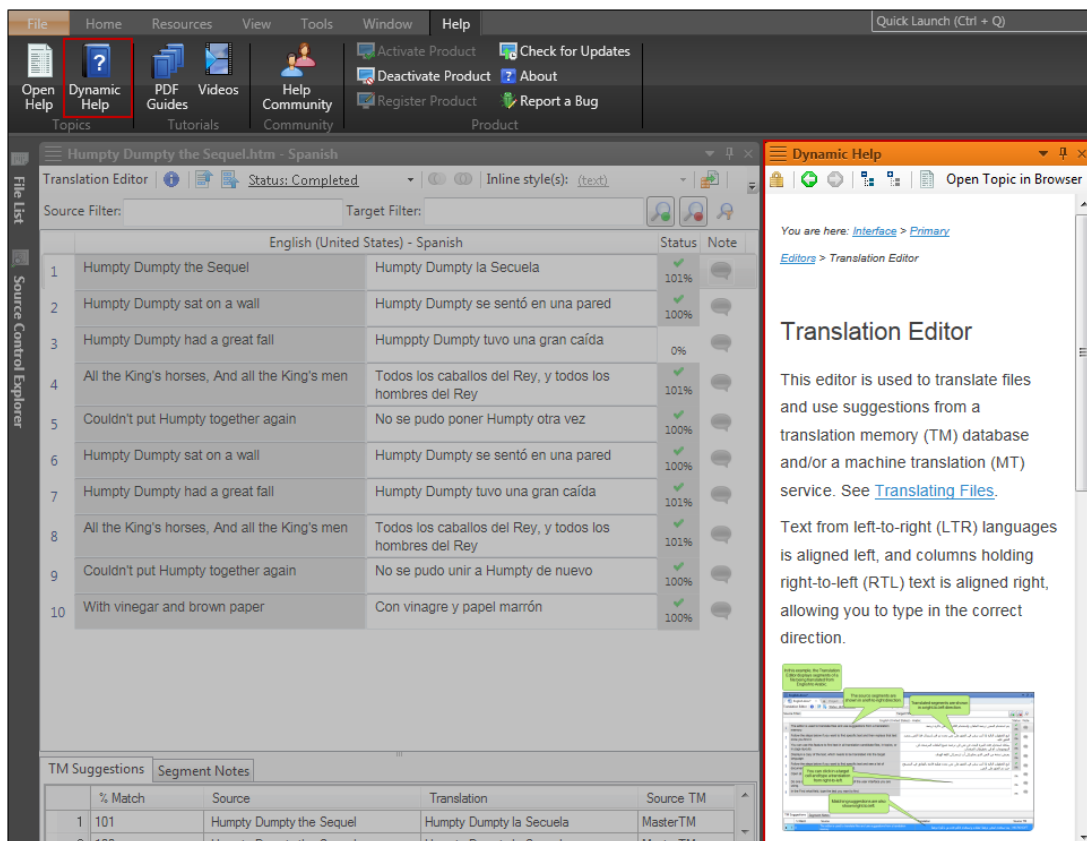
You can also open the full online Help in your browser. To do this, you can click the **Open Topic in Browser** button at the top of the dynamic Help window pane.

![Dynamic Help panel showing the Translation Editor topic with "Open Topic in Browser" button highlighted]

Dynamic Help

Open Topic in Browser

You are here: *Interface* > *Primary*

Editors > *Translation Editor*

Translation Editor

This editor is used to translate files and use suggestions from a translation memory (TM) database and/or a machine translation (MT) service. See Translating Files.

Text from left-to-right (LTR) languages is aligned left, and columns holding right-to-left (RTL) text is aligned right, allowing you to type in the correct direction.

You can also select **Help>Open Help**.
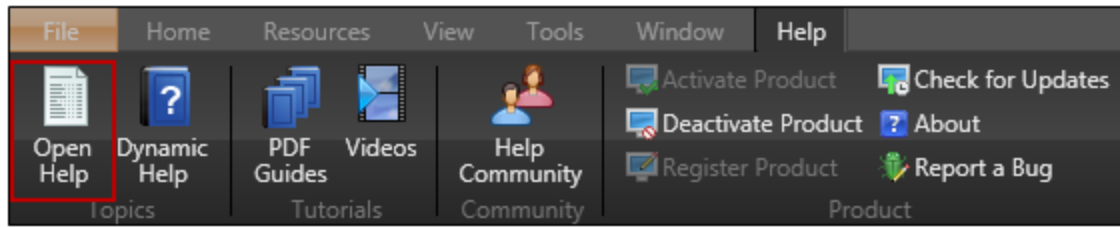
Local Help System

Because the Help system is hosted online, you may not be able to access it if you are behind a fire-wall or if you have a slow connection. In that case, you can use a local version of the Help. Lingo will even switch to the local Help automatically if it detects that you cannot connect to the external Help system.

If you want to use the local Help manually, open the Options dialog (**File > Options**), and on the **General** tab select **Prefer Local Help**.

> NOTE: The local Help system is only updated when the application is updated. If you want to be sure that you have the most current version of the Help, disable the local Help.

Image Replacement

In previous versions of Lingo, if you wanted to replace images in your project, you would need to replace them individually, or you could replace them in you original Flare project and then update your Lingo project. New methods to replace images have been added to Lingo 10. You can now replace several images at once using a batch. This method also supports multilingual projects. You can also replace images in skins.

Batch and Multi-Language Image Replacement

You can replace an image in Lingo instead of deleting and adding the image manually. You can also create a batch that lets you replace many images at the same time, without needing to retranslate their callouts.

If you are working in a multilingual project, you can replace the original images with different images for each language in your project. This is very useful if you need to localize images for different languages or countries.

To open the Batch Replace Images dialog, select **Home>Batch Replace Images**, or select in the File List's local toolbar. The dialog that opens, you can select the folder containing the original images and sync them to a folder containing replacement images. If the images in the replacement folder have the same names as the images in the original folder, the replacement images will automatically be linked to the original images. You can also manually replace images by clicking the link in an image's cell. If you click on the original or replacement image in the grid, you can view a preview of the image at the bottom of the dialog.
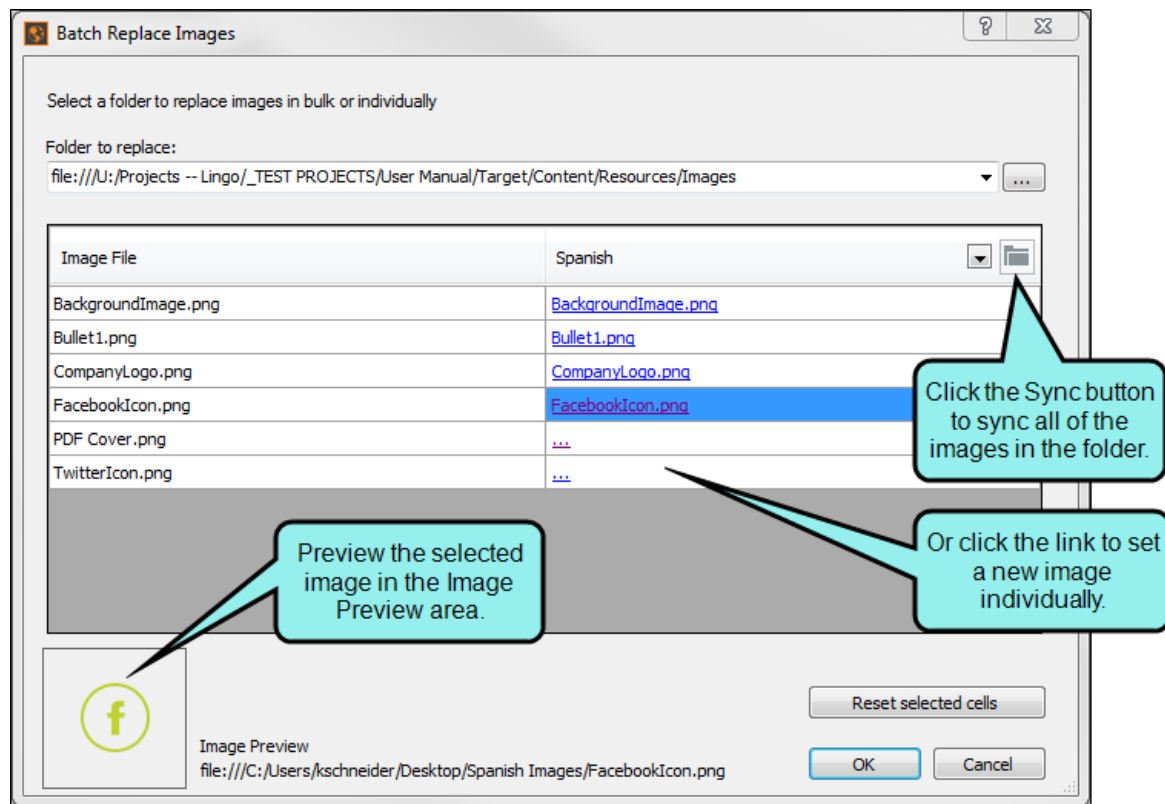
Batch image replacement supports multilingual projects. The currently selected active language is shown in the dialog by default. To replace images for a different language in your project, select a new language from the drop-down at the top of the language column.
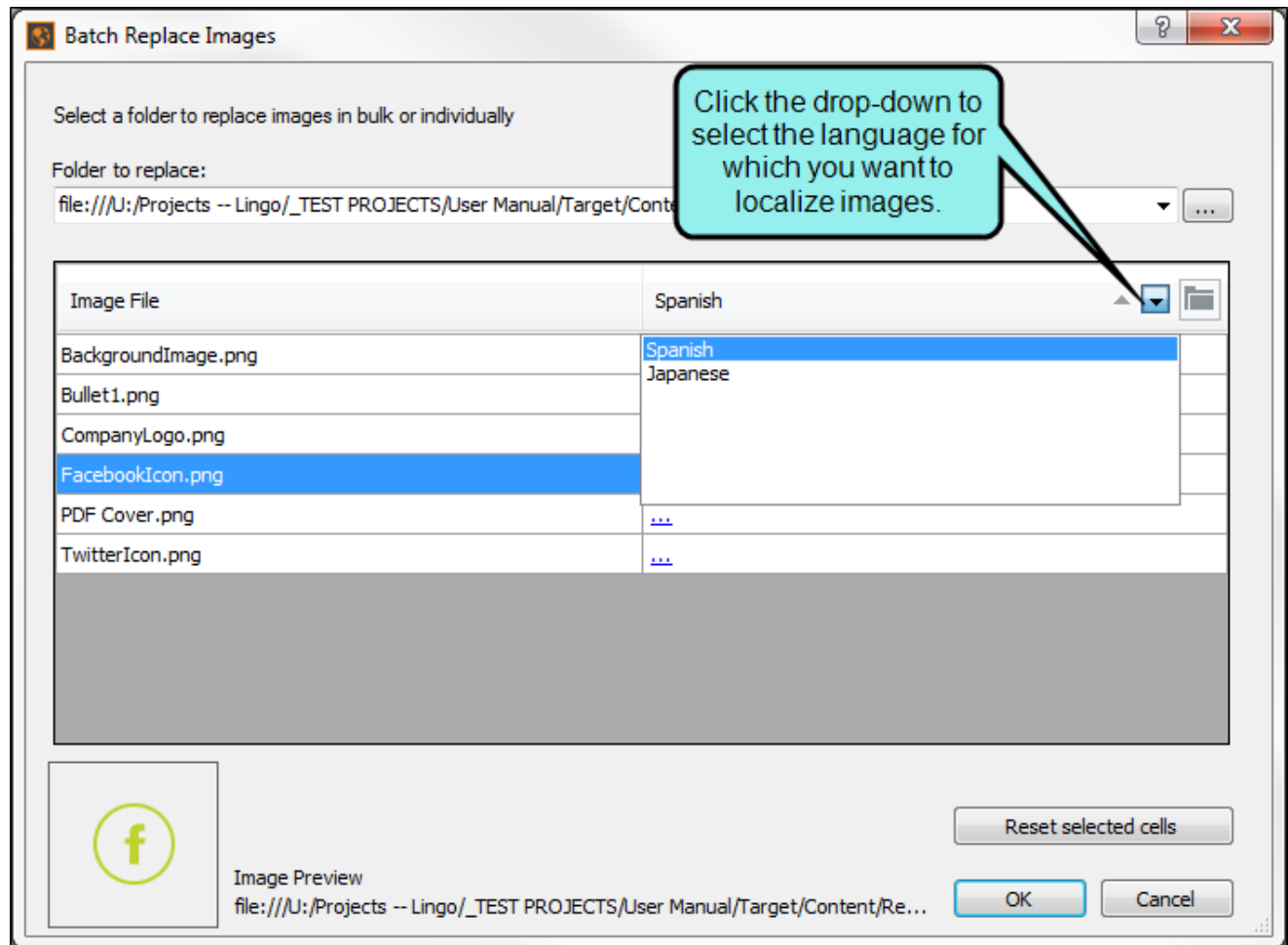
When you click OK, the original images will be replaced with the new images that you selected. Any translated callouts that were part of the original images will be transferred to the replacement images.

☆ EXAMPLE

You are translating documentation for a company whose products are used in both English- and Spanish-speaking countries. You need to localize the images in the documentation so the text on the images is appropriate for the language used in the documentation. To do this, you can use the Batch Image Replace dialog to sync the images from the English source project with the localized Spanish images.
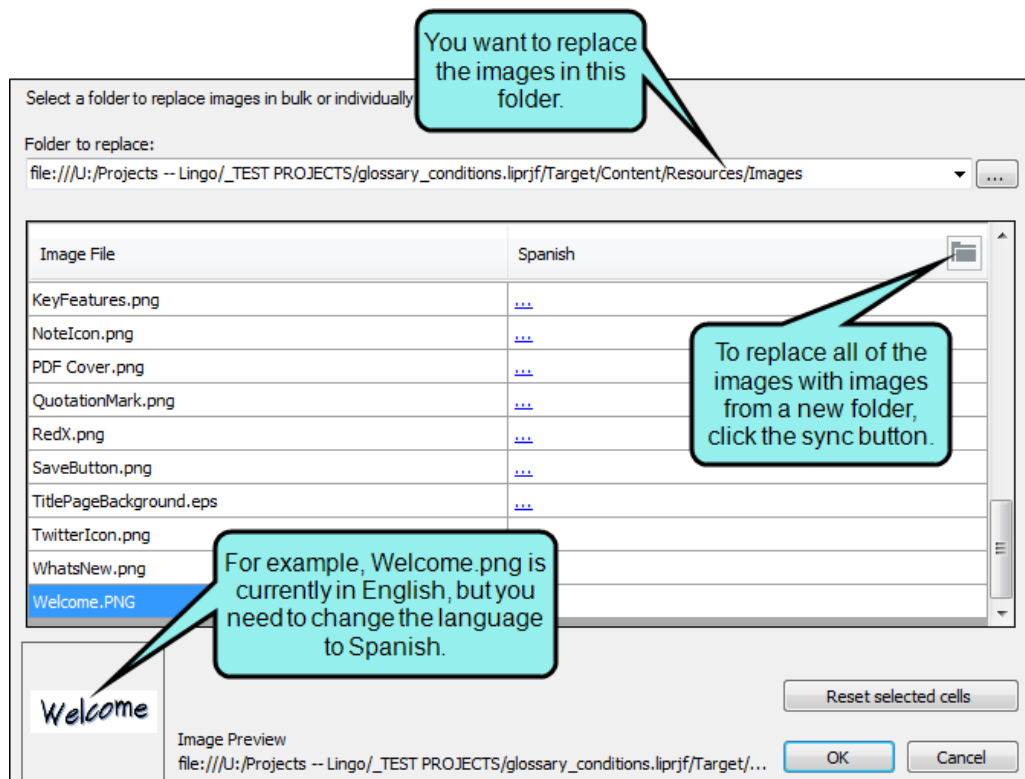
To do this, you create a folder with the localized Spanish images. After you click the button in the Batch Replace Images dialog, you select the Spanish Images folder.
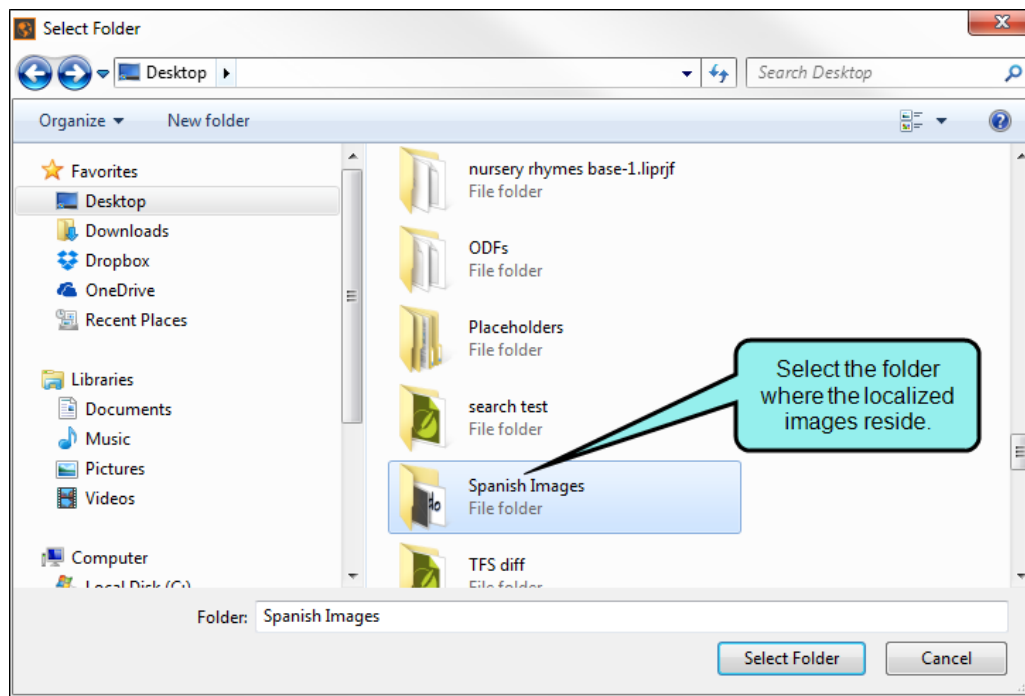
⭐ Any images that have identical file names to those in the original Images folder are automatically replaced with your localized Spanish images.
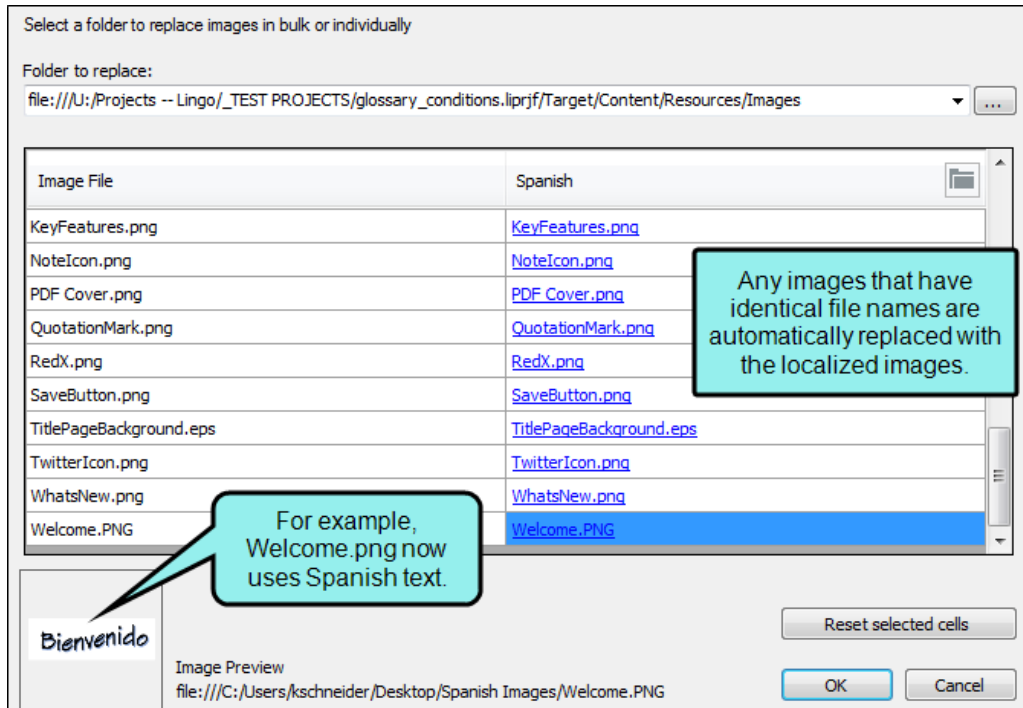
NOTE: If there are not any images in your project, batch image replacement is disabled.

NOTE: Callouts from the original image are transferred to the replacement image and can be translated in Lingo. However, if your replacement image *also* contains callouts, these callouts will be flattened during the replacement process, and will no longer be editable. As a best practice, choose replacement images that do not include callouts.

NOTE: When performing a batch replacement, you can replace images in subfolders. To sync correctly, the subfolders must have the same names in the original folder and the replacement folder. Folders with different names will not sync.

NOTE: If you have localized images and you update your Lingo project, a confirmation dialog will ask you if you want to replace the localized images with the images from the source project.

NOTE: If you are translating a MadCap Mimic movie, you can replace frame background images. However, image objects are not replaceable.

NOTE: If you localize an image and then reopen the Batch Image Replacement dialog, the replacement link next to the original image will say "localized." This alerts you that you have already replaced that image with a new image. However, you can still click the link and replace the image with a new image.

Skin Image Replacement

In addition to replacing images in topics, you can also replace images used in skins. You can replace logos in Flare's HTML5 skins, and you can replace the About box in Mobile and WebHelp skins.

To replace the skin image, open the skin in the Translation Editor, then click ⬚ in the local toolbar. The Replace Skin Images dialog opens. You can see a preview of the current skin image in the dialog. Click **Browse** to select a new image.

☆ **EXAMPLE**

You need to localize hundreds of images in a Flare project. You have already used the Batch Image Replace dialog to localize images in the topic files, but you also need to localize the skin files, so the online Help will show the correct version of your company logo. To do this, you set the current target language in the Project Properties dialog...
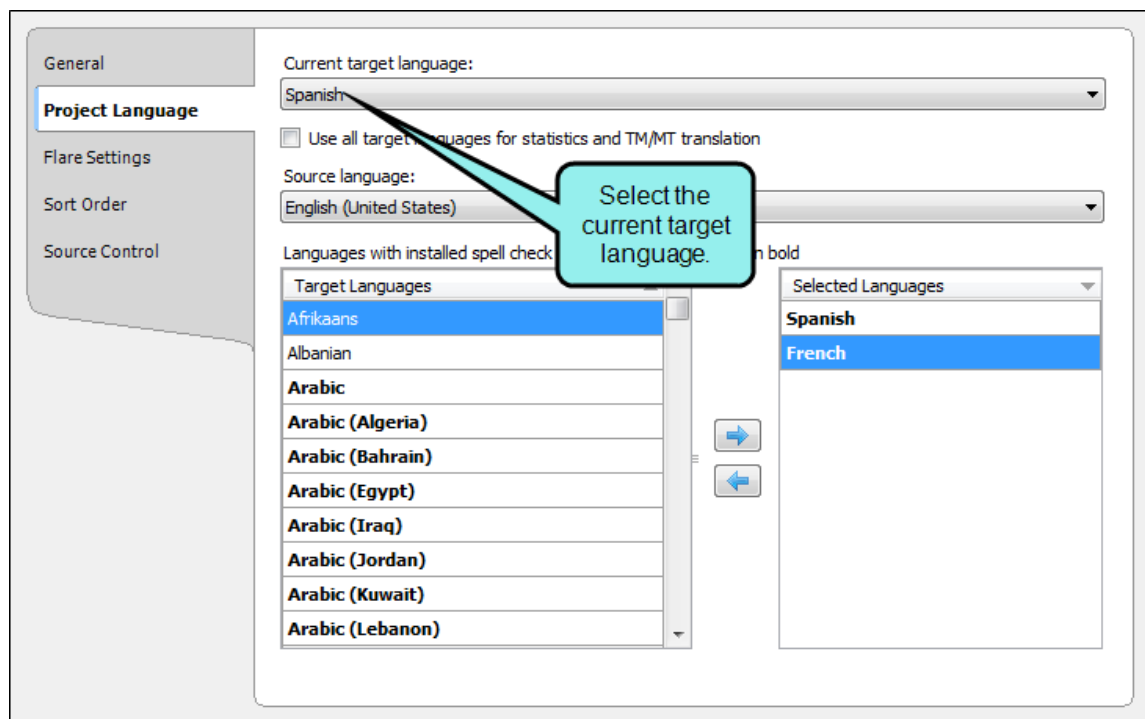
⭐ ...then you open the skin file for that language. In the local toolbar, select **Replace Skin Images**.
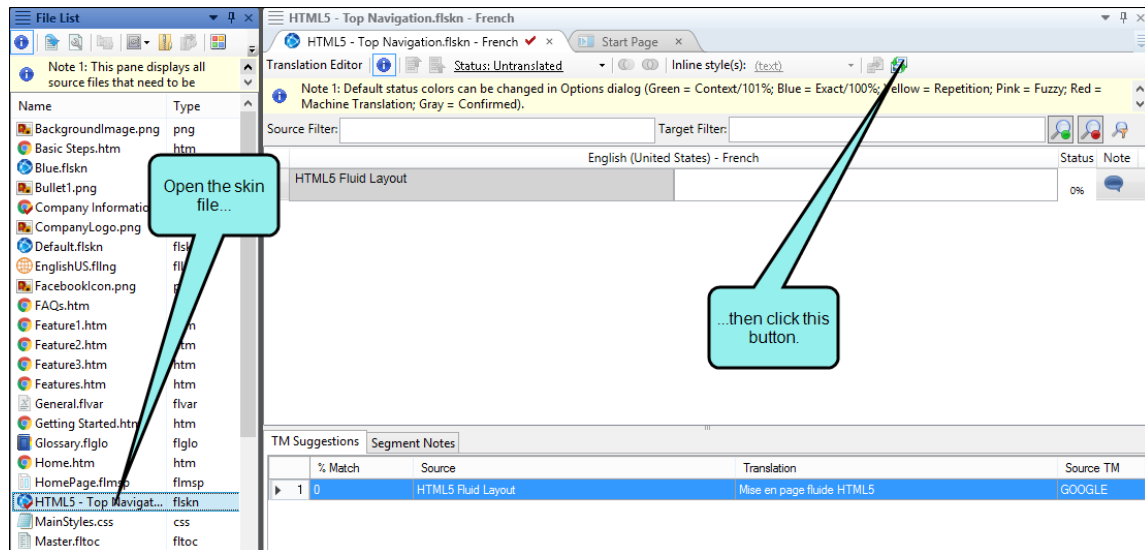
In the dialog that opens, replace the current logo...

☆ ...with the new logo.
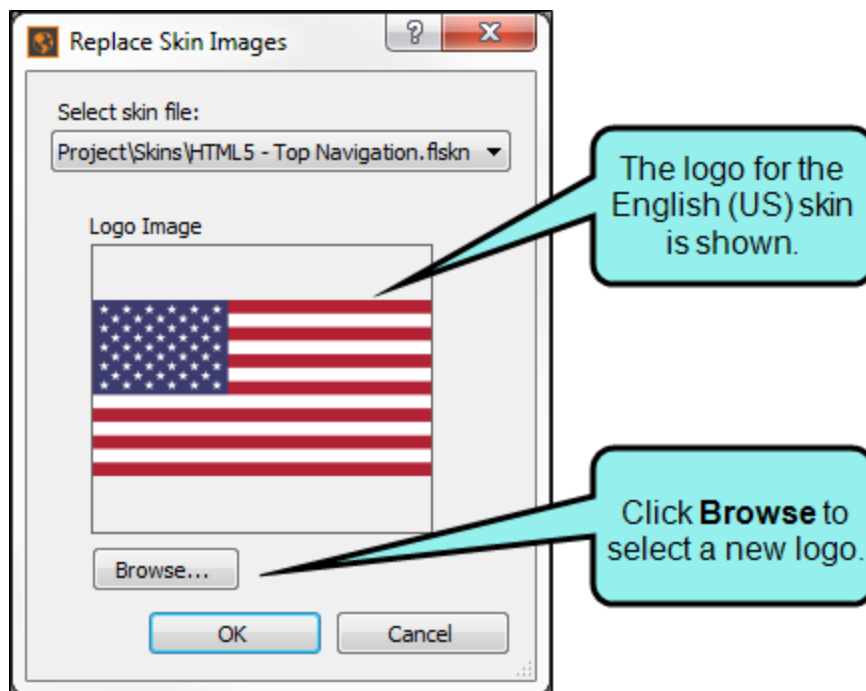
Since the target language is Spanish, we selected the Spanish flag to use as the logo.
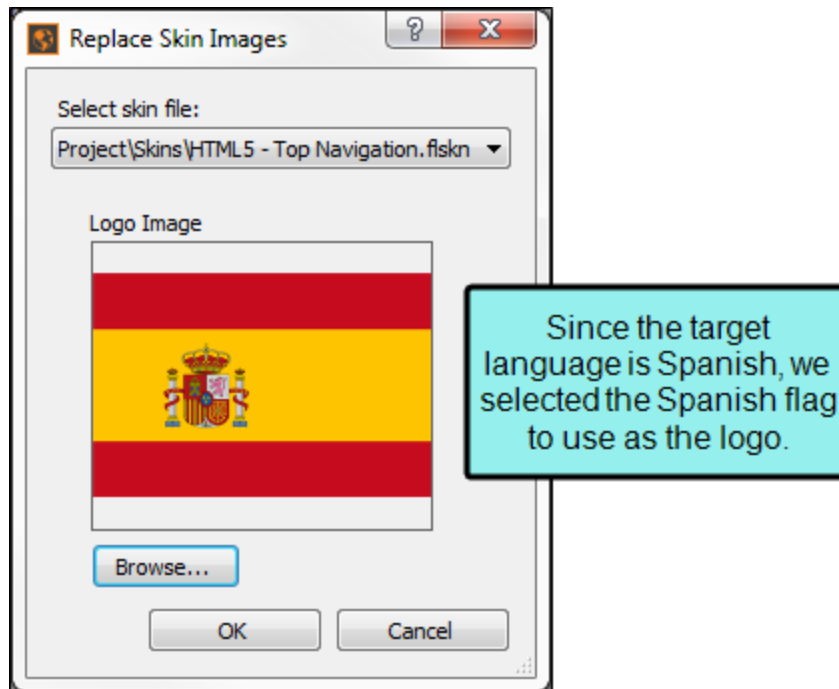
☆ You do this for each target language. When the project is exported back to Flare and the file outputs are generated, the online Help shows the correct logo for each skin.
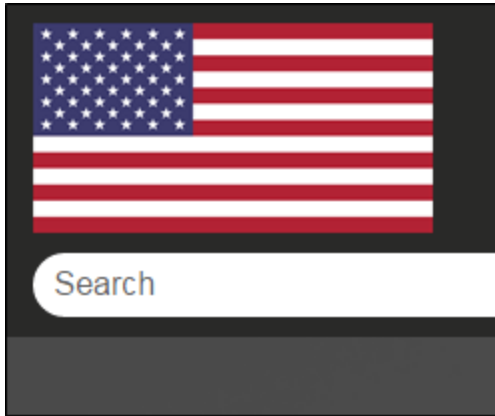
> **NOTE:** You can replace skin images with .gif, .jpeg, .jpg, and .png image types.

> **NOTE:** If you need to localize images in your skins, you can select a different target language for the project from the Project Properties dialog (**File>Project Properties**). Then for that language, open the skin file(s).

Project Export Enhancements

Project export has been updated so Lingo will update previously exported versions of the project. In previous versions of Lingo, if you wanted to re-export an existing project, you would need to create a second copy of the project, or delete the previous export.

When you update a project in Lingo 10, a confirmation dialog will alert you if the export destination already exists. When exporting the project to an existing location, only changed and new files will be exported.
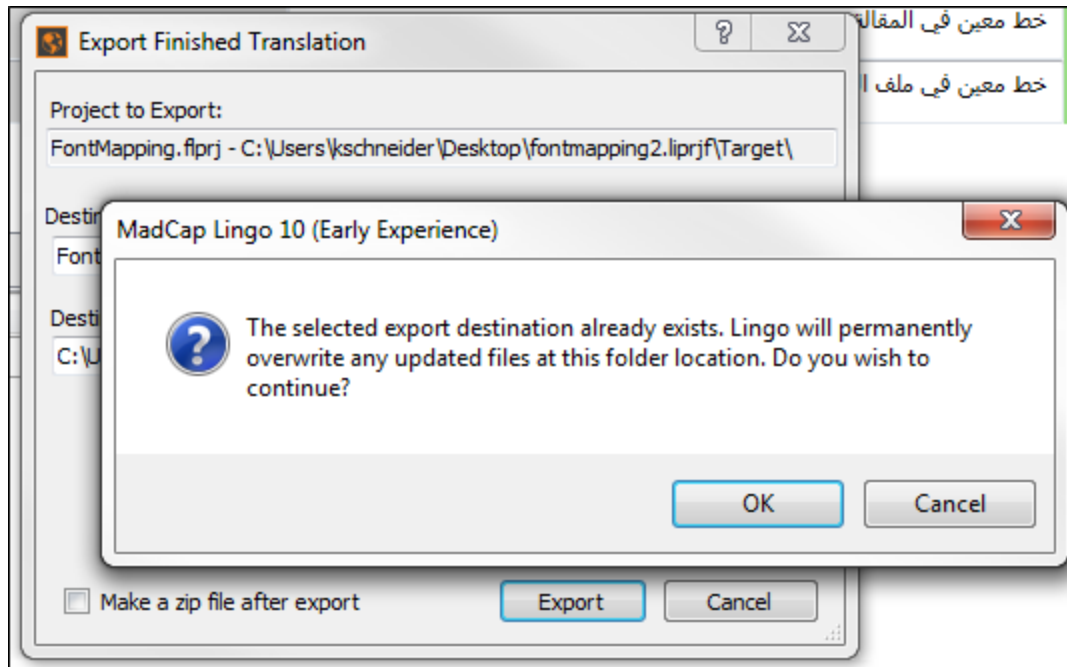
You can view saved export logs in the Project Properties dialog. If an export fails, these logs provide information about what caused the failure. You can save and view these logs.

Project Properties Dialog

In previous versions, you could view and change basic project information using the Project Editor. You can now access these and other project-related settings using the Project Properties dialog.

The Project Properties dialog organizes project settings and information into the following tabs:

- **General** View the project's name, folder, domain, client, and log files. You can also open the Lingo project folder.

- **Project Language** View and change the project's current target language and the source file's language. You can also add and remove languages from the project and select the language (s) to use when compiling statistics and applying the translation memory and machine translations.

General	Current target language:
Project Language	Spanish
Flare Settings	☐ Use all target languages for statistics and TM/MT translation
Sort Order	Source language:
Source Control	English (United States)

Languages with installed spell check dictionaries are displayed in bold

Target Languages		Selected Languages
Afrikaans		**Spanish**
Albanian		**Finnish**
Arabic (Algeria)	→	**Arabic**
Arabic (Bahrain)	←	
Arabic (Egypt)		
Arabic (Iraq)		
Arabic (Jordan)		
Arabic (Kuwait)		
Arabic (Lebanon)		
Arabic (Libya)		

- **Flare Settings** Select the variable sets or individual variables you want to flatten, select project filtering options, and choose whether or not you want to convert inline snippets to text. This tab is only available in Flare projects.
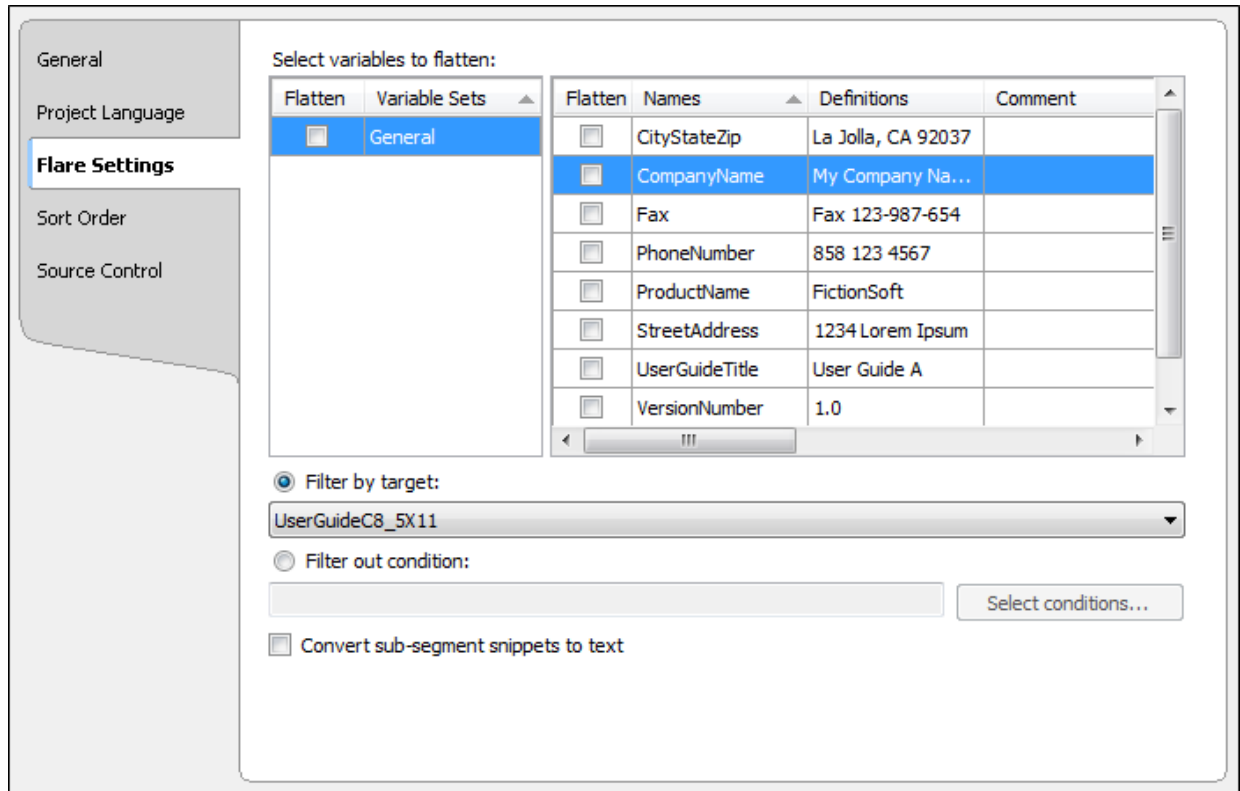
- **DITA Settings** Select conditional attributes to filter a DITA project. This tab is only available in DITA projects.

- **Sort Order** Add "Sort As" entries for index and glossary terms. These terms will be used when you export the project back to Flare to create a custom index and glossary sort order. This tab is only available in Flare projects.

- **Source Control** Select options related to source control, such as binding and unbinding a project. If the project is already bound, you can use the source control button to perform tasks on files in your project.

Report Enhancements

Improvements have been made to Lingo's reporting features. These include new quality assurance (QA) reports and report styling updates to improve readability.

New Quality Assurance Reports

Two new quality assurance reports have been added in Lingo 10:

- **Localized image files** This report shows the localization status of all of the images in your project. The report shows the languages for which you have chosen a new image ("localized" images), as well as those still using the original source file ("unlocalized" images).

> ☆ EXAMPLE
>
> Let's say you are translating a project into 10 different languages. As part of the translation process, you need to localize all of the images in the project. During the course of your project, several new topics are added to the source project, and you receive new images that you need to localize.
>
> Because there have been so many changes to the project, you periodically run the localized image files report to see the localization status of each image in the project. When you run the report, you can see each original image, what languages have been localized for each image, and what languages have not been localized for each image. This helps you keep track of which images you still need to update.

☆ The following image shows how such a report might look.

- **Translations that match machine translation** This report compares the translated target segment with any available machine translations (MTs). This allows you to review your translations against the machine translations and make changes if necessary. You can change the threshold for the minimum MT match percentage, and you can set a minimum number of words in the segment (to prevent obvious matches, such as names or short phrases, from appearing in the results).

> NOTE: This report is unavailable if an MT provider is not connected to the project.

☆ EXAMPLE

Let's say you regularly send your Lingo projects out for translation. You want to make sure that your translators are not relying too heavily on MT, because your company uses several company-specific terms that are located in your concordance and in your translation memories, but would not be found in a translation from an MT.

When you get your translations back from the translators, you run a QA report to see which translations match the MT. You set the percent match at 90% and the minimum word match to 4 words.

When you look at the report, you can see which translations are at least a 90% match to the MT. You may want to review these translations to make sure that your translators made the appropriate adjustments to the machine-generated translations.

Report: Machine translation target similarity

Location	Source Segment	Target Segment	MT Segment	MT Provider	MT Match Percent
Project Total					72%
Humpty Dumpty the Sequel.htm - Spanish Segment #1	Humpty Dumpty the Sequel	Humpty Dumpty el Sequel	Humpty Dumpty el Sequel	GOOGLE	100%
Humpty Dumpty the Sequel.htm - Spanish Segment #2	Humpty Dumpty sat on a wall	Humpty Dumpty sentado en una pared	Humpty Dumpty sentado en una pared	GOOGLE	100%
Humpty Dumpty the Sequel.htm - Spanish Segment #6	Humpty Dumpty sat on a wall	Humpty Dumpty se sentado en una pared	Humpty Dumpty sentado en una pared	GOOGLE	91%
Humpty ... Spanish ...	... Humpty Dumpty.	Mi nombre es Dumpty.		GOOGLE	100%
Jack and ... Segment ...	... Jill went up	Jack y Jill sub... colina		GOOGLE	93%
Jack_and_Jill.jpg - Spanish Segment #1	My name is Jill.	Mi nombre es Jill.	Mi nombre es Jill.	GOOGLE	100%
Jack_and_Jill.jpg - Spanish Segment #2	My name is Jack.	Mi nombre es Jack.	Mi nombre es Jack.	GOOGLE	100%
Little_Miss_Muffet.jpg - Spanish Segment #1	My name is Little Miss Muffet.	Mi nombre es Little Miss Muffet.	Mi nombre es Little Miss Muffet.	GOOGLE	100%
MyGlossary.flglo - Spanish Segment #2	A low seat or a stool to rest the feet of a seated person.	Un asiento bajo o un taburete para descansar los pies de una persona sentada.	Un asiento bajo o un taburete para descansar los pies de una persona sentada.	GOOGLE	100%

This report shows the target segments and their corresponding MT translations.

You can also see the MT match percentage for each segment, and also for the project as a whole.

Report Styling Updates

Lingo's quality assurance (QA) and statistics reports have been restyled with spacing and formatting improvements so they are easier to read.

QA REPORTS

Lingo's QA report has been updated to use better spacing and alternating row colors to improve readability. The location of the target segment has been moved to the first column so it is the first thing you see. If you run multiple QA reports, lines help to separate each report.

Lingo 9

Mismatching punctuation in source and target for Nursery Rhymes New.liprjf

Source Segment	Target Segment Variant	Location
Couldn't put Humpty together again	No se pudo poner Humpty otra vez!	Humpty Dumpty.htm - Spanish Segment #5
With vinegar and olive paper	Con el vinagre y papel olivo.	Jack and Jill.htm - Spanish Segment #9
"Little Miss Muffet" is a nursery rhyme, one of the most commonly printed in the mid-twentieth century.	Little Miss Muffet is a nursery rhyme, one of the most commonly printed in the mid-twentieth century.	Little Miss Muffet.htm - Spanish Segment #2

Lingo 10

Report: Mismatching punctuation in source and target

Location	Source Segment	Target Segment
Humpty Dumpty.htm - Spanish Segment #5	Couldn't put Humpty together again	No se pudo poner Humpty otra vez!
Jack and Jill.htm - Spanish Segment #9	With vinegar and olive paper	Con el vinagre y papel olivo.
Little Miss Muffet.htm - Spanish Segment #2	"Little Miss Muffet" is a nursery rhyme, one of the most commonly printed in the mid-twentieth century.	Little Miss Muffet is a nursery rhyme, one of the most commonly printed in the mid-twentieth century.

STATISTICS REPORTS

Lingo's statistics report has been updated to use more defined tables with clearer spacing. Each table uses grid lines to provide visual distinction between each row, and colors help you quickly identify the match percentages in the table. Lines help to separate individual statistics reports for each topic in the project.

Lingo 9

Statistics for Statistics Reports.liprj

Match Types	Segments	Words	Characters
Duplicates	91	201	1586
Translated	31	200	838
101%	10	108	588
100%	12	52	225
95% - 99%	2	9	58
85% - 94%	1	3	11
75% - 84%	2	5	17
50% - 74%	7	20	121
No Match	98	320	1970
Total	254	918	5414

Lingo 10

Statistics for Statistics Reports.liprj

Match Types	Segments	Words	Characters
Duplicates	72	163	1548
Translated	31	200	838
101%	3	12	61
100%	19	148	752
95% - 99%	2	9	58
85% - 94%	1	3	11
75% - 84%	2	5	17
50% - 74%	7	20	121
No Match	97	318	1968
Total	234	878	5374

Smart Quotes

Improvements have been made to the way Lingo handles smart quotes when translating a segment. Previously, Lingo would automatically replace straight quotes with curly quotes, regardless of the language being translated. However, Lingo now replaces quotes based on your project's current target language.

1	"Smart quotes in English"	"Smart quotes in English"
2	«Smart quotes in French»	«Des guillemets en Français»
3	„Smart quotes in Danish"	„Krøllede anførselstegn i dansk"

> NOTE: Selecting the smart quote option does not automatically replace all of the straight quotes that you already have in your project. Likewise, existing smart quotes are not automatically replaced with language-specific smart quotes. If you need to update the quotes in your project, It is recommended that you use the Find and Replace in Files window pane to find all of the files in the project containing straight quotes (or existing smart quotes) and then enter smart quotes in their place.

TOC View Window Pane

You can now filter the topics in your Lingo project using the TOC View window pane. This view allows you to filter the topics in your project by any table of contents (TOC) in the source Flare project. This is especially useful if you are working with a very large Lingo project, because not only does it let you filter by topic files and by TOC, but it also lets you see the order of the topics and the way the topics fit together in the final output.

When in TOC View, you can also view topics' conditional indicators, save linked files for review, and create or merge translation bundles.

⭐ **EXAMPLE**

You are working with a large Lingo project. It includes many different file types: images, skins, page layouts, and topics. There are several TOCs in the source Flare project, and different reviewers are responsible for the content that belongs in each TOC. You need to send the appropriate review packages to each reviewer.

You open the TOC View window pane and filter the files to show a specific table of contents. This lets you see only the topic files that belong in that TOC.
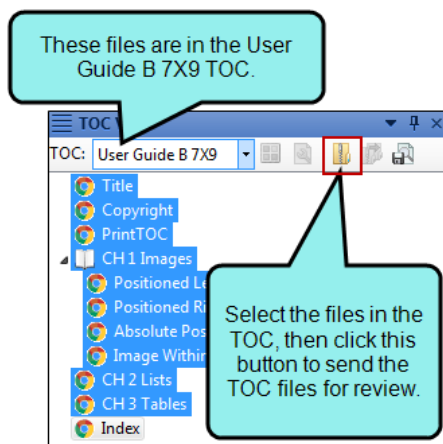
☆ Then you can create a review package and send the topics in that TOC to the reviewer.

Create Bundle

Bundle Settings
- ◉ Lingo Project Bundle
- ○ XLIFF Files Bundle
- ☐ Include Project Translation Memories
- ☐ Include Project Termbases

All of the files that you selected from the TOC are already included in the bundle.

Bundle Files

Project Files Filters:

| All File Types ▼ | All Translation Statuses ▼ | All Difference Status ▼ |

Project Files

Name	Path
Appendix.htm	Content\Print O...
Basic Steps.htm	Content\A_Getti..
Chapters7X9.flpgl	Content\Resour...
Chapters8_5X11...	Content\Resour...
ChaptersLandsc...	Content\Resour...
ChaptersLandsc...	Content\Resour...
Company Inform...	Content\E_Foot...
Conditions.htm	Content\C_Con...
ConditionsNot...	Content\C_Con...
FAQs.htm	Content\E_Foot...
General.flvar	...\Variable...

Output Files

Name	Path
Title.htm	Content\Print O...
Copyright.htm	Content\Print O...
TOC.htm	Content\Print O...
Images.htm	Content\D_Imag...
ImagePositioned...	Content\D_Imag...
ImagePositioned...	Content\D_Imag...
ImageWithAbsol...	Content\D_Imag...
ImageWithinList....	Content\D_Imag...
Lists.htm	Content\B_Links...
Tables.htm	Content\C_Con...
Index.htm	Content\Print O...

If you need to include additional files, you can select them and add them to the bundle.

OK Cancel

☆ Later, when the reviewer sends their review package back to you, you can merge the review package back into Lingo. If you want, you can do this from TOC View, as well.
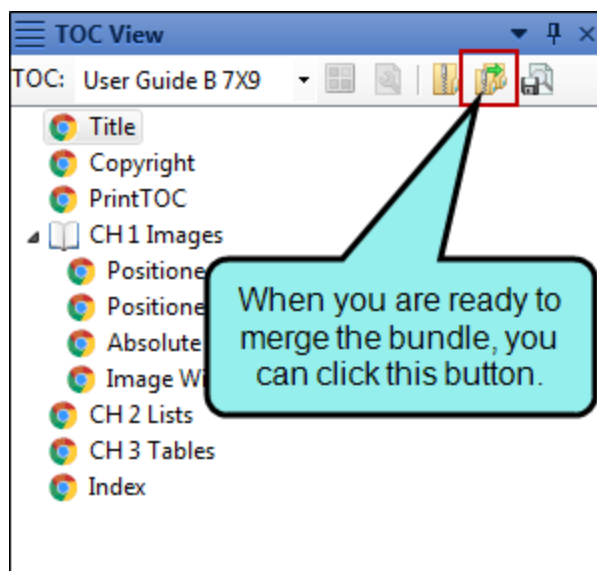
☆ When you are ready to create a bundle for your another translator, select a different TOC from the TOC drop-down. The files in the list update, and you can prepare a new bundle.
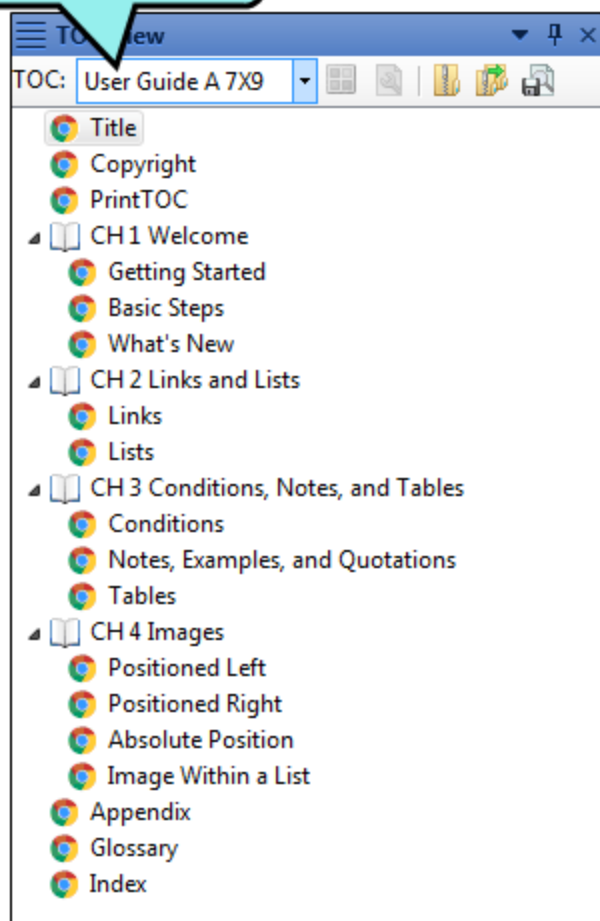
Translation Editor Enhancements

Improvements have been made to how the Translation Editor handles conditional text and concept and index tags. Condition tags are now supported on block level and inline text, and also on glossary and index terms. Additionally, concept and index tags, which have always been supported, have been cleaned up to improve usability in the Translation Editor.

Improved Conditional Text Support

Condition tag support has been enhanced in Lingo 10. In previous versions, conditional text was only supported on files. Lingo 10 also supports block level and inline conditional text, as well as condition tags on glossary terms, snippets, and CSS styles.

When you are working with segments in the Translation Editor, you will see yellow condition tags on either side of the content that is conditioned. Translators can drag these tags to the appropriate location in the target segment to indicate where the conditional text indicator should appear in the exported project.

Additionally, you can now filter content using condition tags. When creating a new Lingo project, you can filter out one or more conditions.

☆ EXAMPLE

The segment below includes a condition tag.
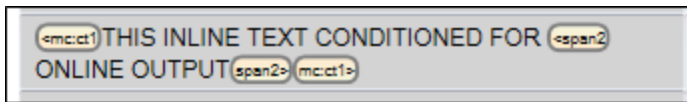
Lingo places yellow conditional text tags on either side of the text that were marked with a condition tag indicator in the source document. You can hover over the tags to see information about the condition that is applied to the text.
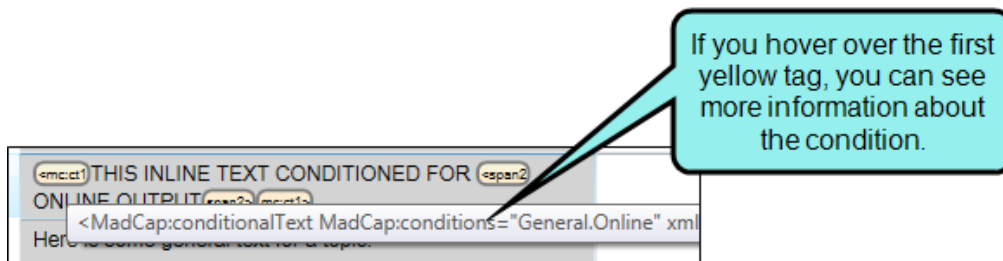
When the translator translates the segments, he can drag the yellow tags into the target segments, so the translation will also include the condition tag. Later, when the translator exports the document back to Flare, the condition tag will appear in the finished project.

⭐ EXAMPLE

You create your glossary in Flare with the intent of using the same glossary file for multiple outputs. To do this, you apply condition tags to each term that indicate which terms belong in which output. Some terms are used in multiple targets, others are used in single targets.

	Condition ▼	Terms	Definition
●	🟡🟥	Batch target	A special target that lets you build and
●	🟥	Block snippet	
●	🟡🟦	Condition tag	
●	🟡🟥🟦	Cross-ref	
●	🟡🟥	Drop-down text	
●	🟡	Footnote	
●	🟦	Single-Sourcing	Reusing content and producing multip
●	🟡🟦	Snippet	A pre-set chunk of content that you ca
●	🟥🟦	Span	A tag that is used to group inline elem
●	🟡🟥	Style	An element to which you assign a cert
●	🟦	Table	A group of intersecting columns and r
●	🟡🟥	Target	One "instance" of an output type. Whe
●	🟦	Text snippet	A snippet that is created out of a porti
●	🟡	Topic	A chunk of information about a partic
●	🟥	Variable	A pre-set term or content that you can
●	🟡🟥🟦	XML Editor	The window in the Flare interface whe

> Your glossary file looks like this in Flare. Each term is marked with one, two, or three condition tags. Each tag is associated with a specific user guide.

Tags		Include	Exclude
General.Online	🟩	☐	☑
General.PDF	🟦	☑	☐
PrintGuides.UserGuideA	🟨	☐	☑
PrintGuides.UserGuideB	🟦	☐	☑
PrintGuides.UserGuideC	🟥	☑	☐

> Your Flare target is conditioned like this.

☆ Later, when you need to send a specific target to your translator, you can filter your Lingo project by the target you need to produce.

☆ When you do this, the condition text rules set in the Flare target are applied to the Lingo project. Because you applied condition tags to your glossary as well as to your topics, when your translator opens the glossary file in your Lingo project, she will only see the terms that need to be translated for that output.

16	Block snippet
17	A snippet that is created out of one or more paragraphs.
18	Span
19	A tag that is used to group inline
20	A span tag doesn't perform any that you apply to inline content.
21	Cross-reference
22	A navigation link that lets you connect text in one topic to another to
23	Cross-references let you create "automated" links that are based on
24	This allows you to keep links consistent and change them in just one
25	Batch target
26	A special target that lets you build and/or publish multiple other targe
27	You can schedule batches to run at any time.

When you import the project, you see the glossary terms and definitions used in the target you selected.

☆ She will also only see the terms for that ouput when she opens the Project Properties
dialog and views the Sort Order information for the glossary.

General	Select file:				
Project Language	Glossary.flglo ▼				
Flare Settings					
Sort Order	Source Entry ▲	Source Sort As	Entry - Spanish	Sort As - Spanish	
Source Control	Batch target				
	Block snippet				
	Cross-reference				
	Drop-down text				
	Span				
	Style				
	Target				
	Variable				
	XML Editor				

☆ EXAMPLE

You use expanding text throughout your Flare project. You do not want your expanding headers to appear in your printed outputs, so you create a style class in your stylesheet that has an OnlineOnly condition tag applied to it, and then set this style class on your expanding headers.
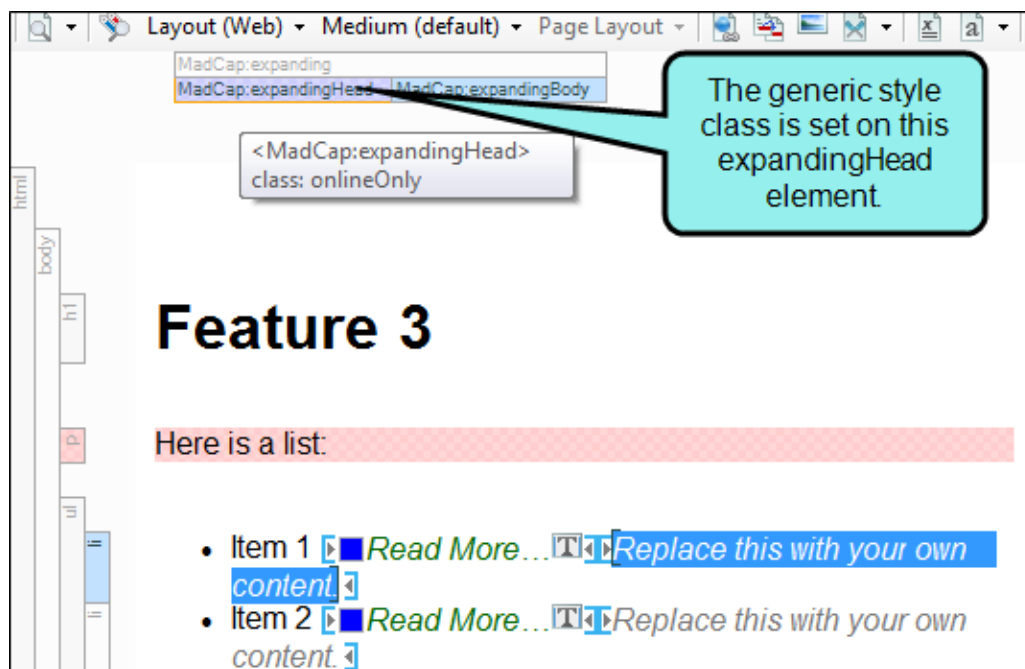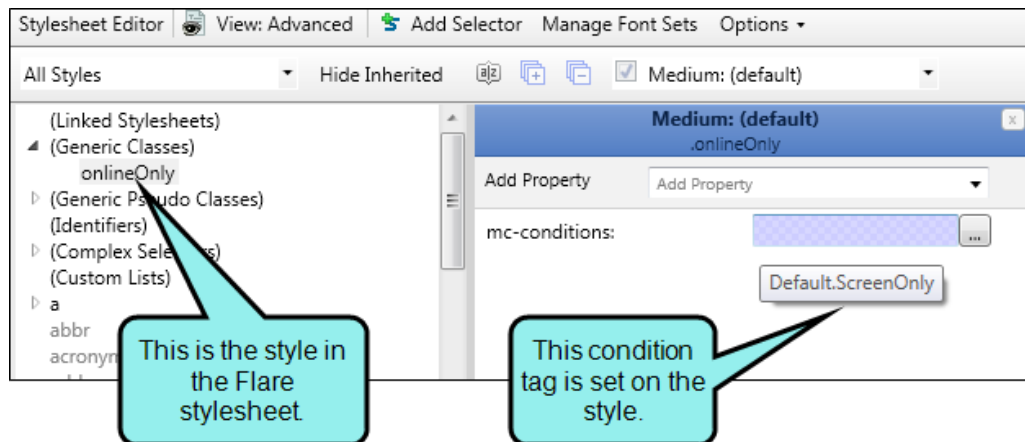
☆ Later, you need to send one of your printed outputs for translation. When you create a Lingo project, you can exclude the OnlineOnly condition tag from the project.
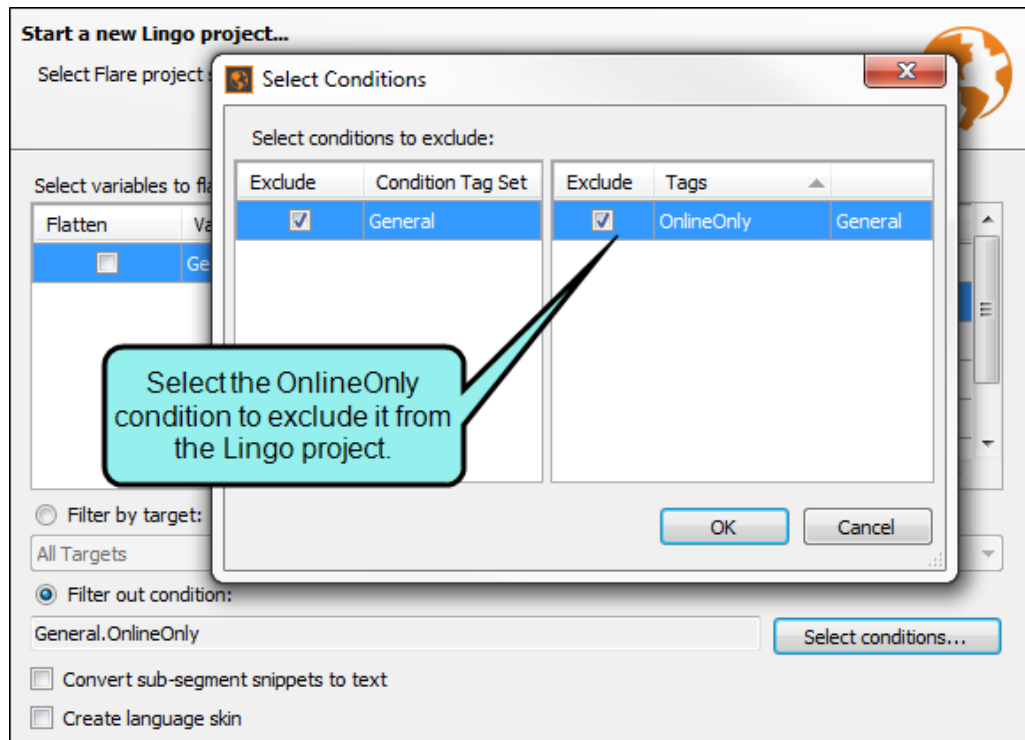
☆ This way, the expanding headers in your online outputs are not included in the content you send to your translator.

The content is excluded and is indicated by a blue tag.
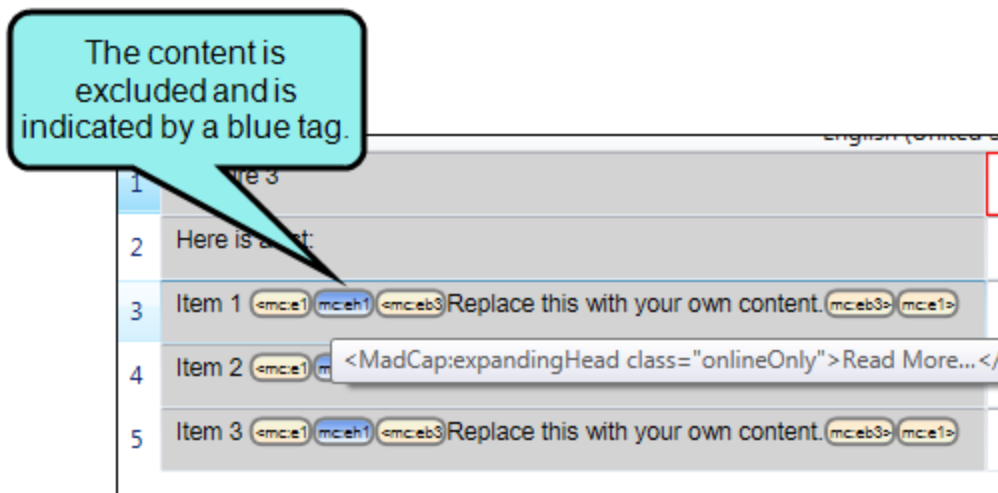
📝 NOTE: Although you can filter out condition tags, Lingo does not explicitly include the condition tags that you have not excluded. For example, if you have condition tags A and B in your project, and you filter out A, you will see content marked with tag B in the project. However, if there is content marked with both tags in your project, it will also be excluded. This is because Lingo excludes anything marked with the A tag, even if the B tag was not also excluded.

If you need to exclude some content and include other content, you can do this by filtering your project using a Flare target with your desired condition filters. Filtering by a Flare target respects both include and exclude condition tag filters.

📝 NOTE: Conditions that "unbind" rather than "exclude" content will not be excluded when you filter out a condition in Lingo. If you notice that content has not been removed but you think that it should be, check to be sure that the condition is not marked as an "unbind" condition in Flare.

Cleaner Index and Concept Tags

In Lingo 10, if an index or concept tag appears at the beginning or end of a segment, it is no longer displayed in the Translation Editor. Removing these tags makes segments look cleaner in the Translation Editor, and prevents translators from needing to move tags that are not related to the segment. Additionally, these tags are no longer displayed in Live Preview mode. Tags are still displayed if they appear between two words.

All tags in a project are translatable when you open their associated file (e.g., index map, concept map) from the File List window pane. This content is added back to the segment when the project is exported.

☆ **EXAMPLE**

When an author writes topics, she includes all of her topic's concept and index tags at the beginning of the topic, by the topic title. That way, when people click an index or related topic link in the output, they are directed to the top of the topic in the online Help.
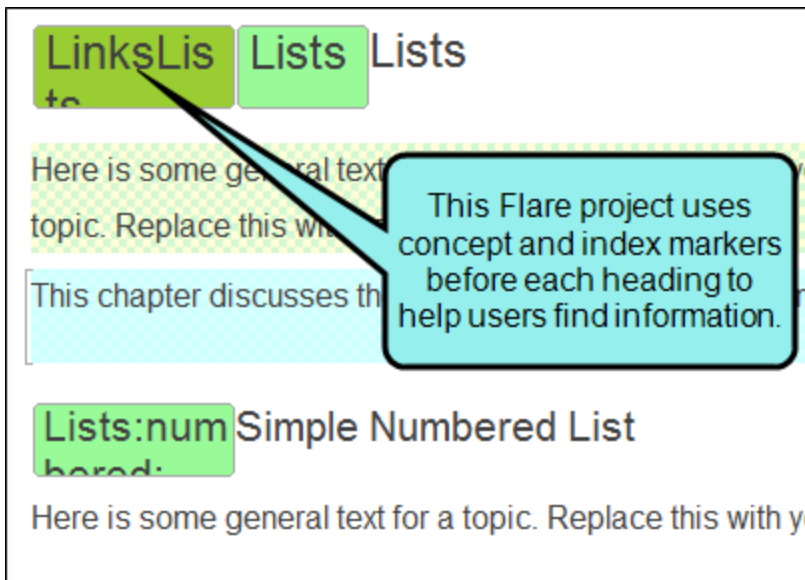
When the author imports her Flare project into Lingo, these tags are automatically removed from the segment. This way the tags do not clutter the beginning of each segment and you can translate just the important content. However, you can still translate these tags when you open the index keyword and concept keyword files in the File List.
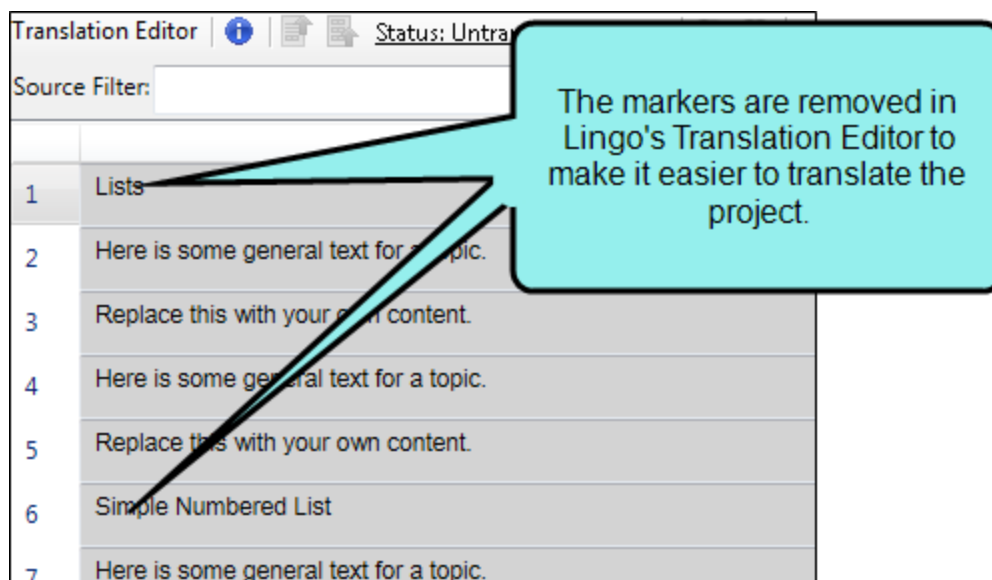
NOTE: Existing translation memories (TMs) that include a translated segment and any type of tag may no longer show a 100% match if they are applied to a segment where the tag has been removed in the Translation Editor. To fix this, update the target segment in the TM to remove the tags.

NOTE: Tags (e.g., spans) are removed from a segment if they wrap the entire translation unit. The exception to this is condition tags, which are *not* removed if they span the entire unit (this is because condition tags can affect translation status).

Similarly, unflattened snippets, conrefs, and variables will be removed from the Translation Editor if they span the entire translation unit (e.g., a snippet that makes up an entire segment).

This content will be added back into the project when you export it. Be sure to translate the content in the associated content file in the File List so it will be translated in the finished project.

New Features for Lingo 10.1

There are also a few new features added in Lingo 10.1.

This chapter discusses the following:

Filter by Multiple Targets

Previously you could select a single Flare target in the Start New Project Wizard when creating a new Lingo project. This allows you to limit the files to only those related to a certain target, rather than importing all of the files from the project.

This feature has been enhanced to let you choose multiple Flare targets in the wizard. When you select the drop-down on this page of the wizard, simply click the check boxes next to the appropriate targets.
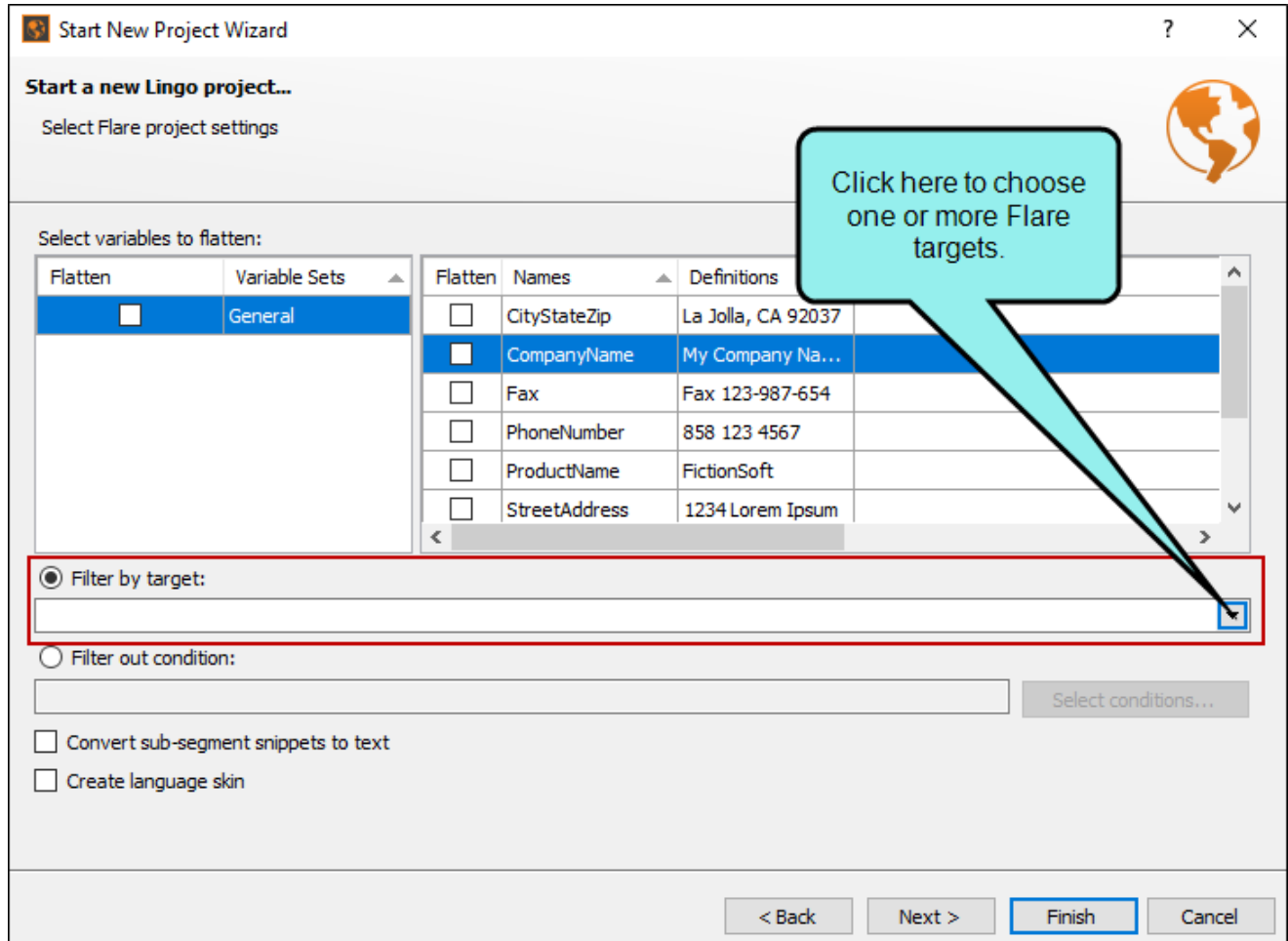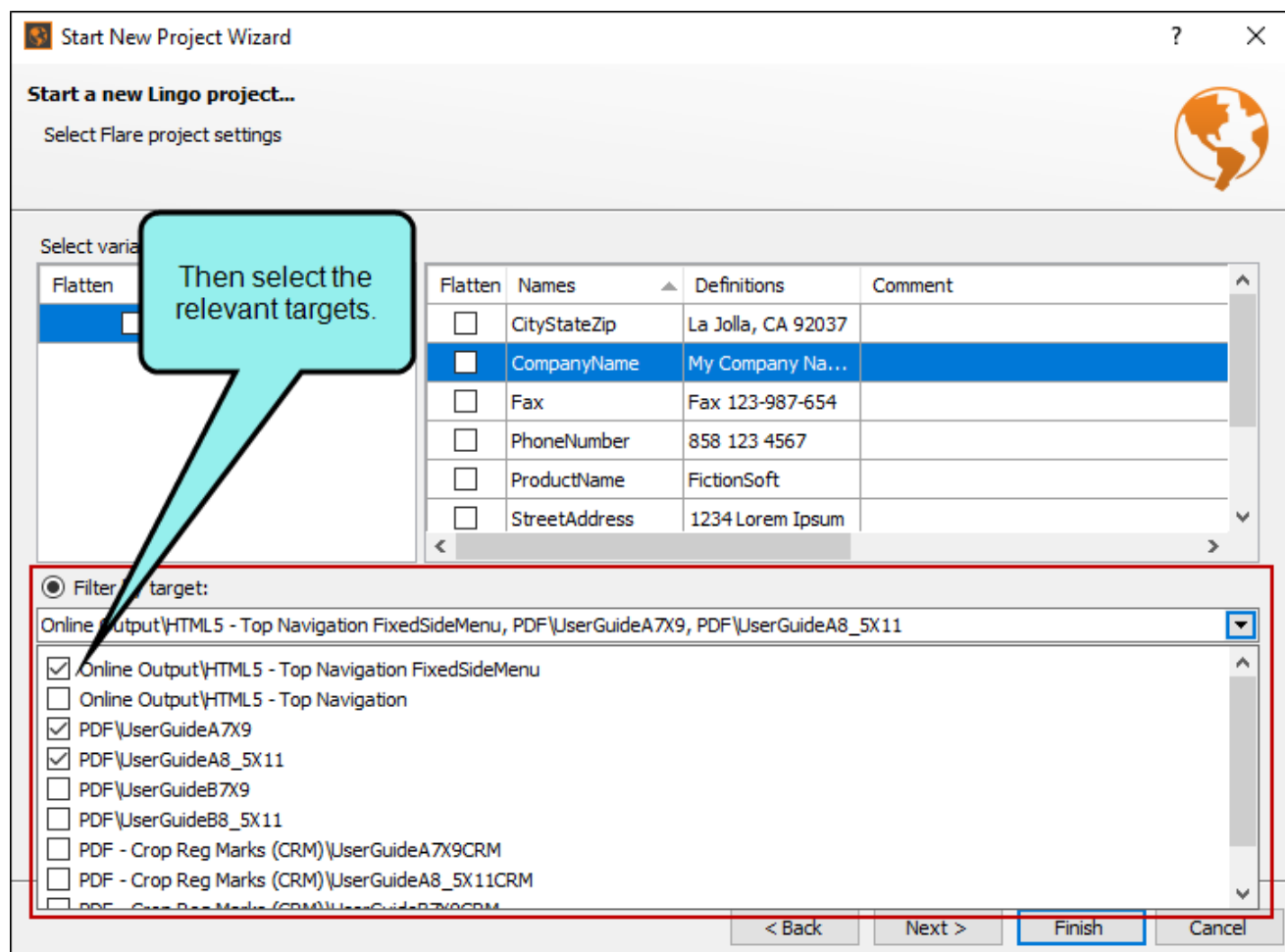
After making your selections, click elsewhere on the page to close the drop-down.

When you finish in the wizard, Lingo will import only the files or segments necessary for those targets.

Show/Hide Spaces in Editors

There is a new space option ¶ in the local toolbar of the Translation Editor and Translation Memory Editor. When you turn this feature on, you can see markers that show where space exists between text in a segment.

The option is currently disabled. Notice that no spacing marks are shown in the grid.

Microsoft Bing Machine Translation Changes

The Microsoft Bing machine translation (MT) API has changed. This means that you must provide the API key when you integrate Lingo with the MT, rather than the user ID and password that were required previously.

In addition, a new field lets your provide a category that you have previously set up in your Microsoft MT account.

Importing Alternate Tag Segments

There is a new field in the Import TMX dialog that lets you import alternate tag segments.
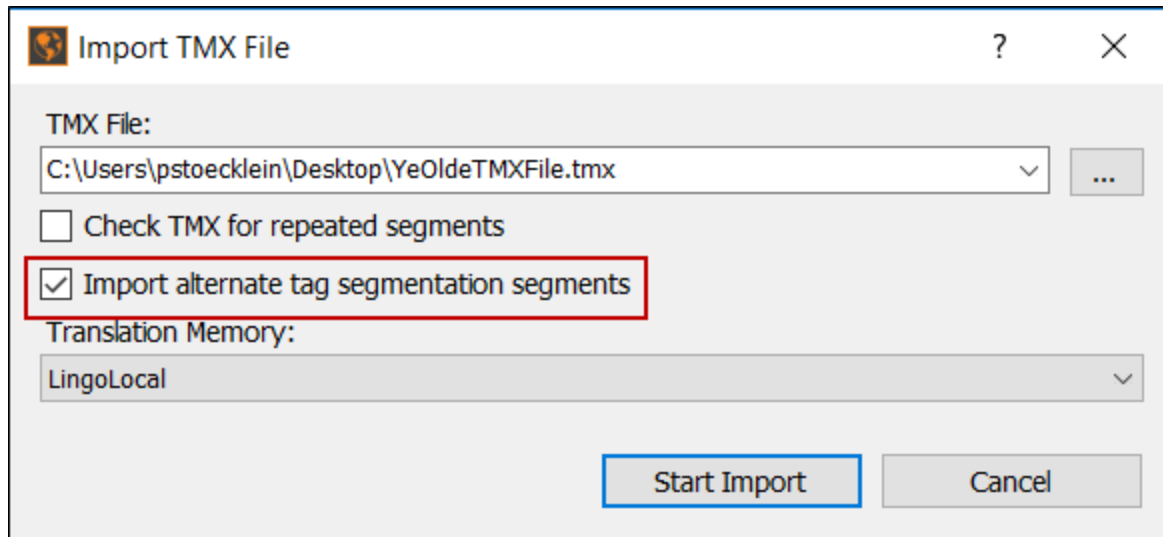
> ☆ **EXAMPLE**
>
> Let's say you are importing a TMX file that was previously used for a third-party translation tool. That tool might have treated the <strong> tag differently than Lingo, so after you import the TMX file you might discover that some match results are not what you expect. By selecting this new option, Lingo will consolidate those tags, which will produce better match results.

New Features for Lingo 10.2

There are also a few new features added in Lingo 10.2.

This chapter discusses the following:

Adobe FrameMaker 2017 Support

If you are translating Adobe FrameMaker files, be aware that Lingo now supports FrameMaker 2017.

Custom Installation

When you install this version of Lingo, you can simply use the default settings, or you can complete a custom installation.
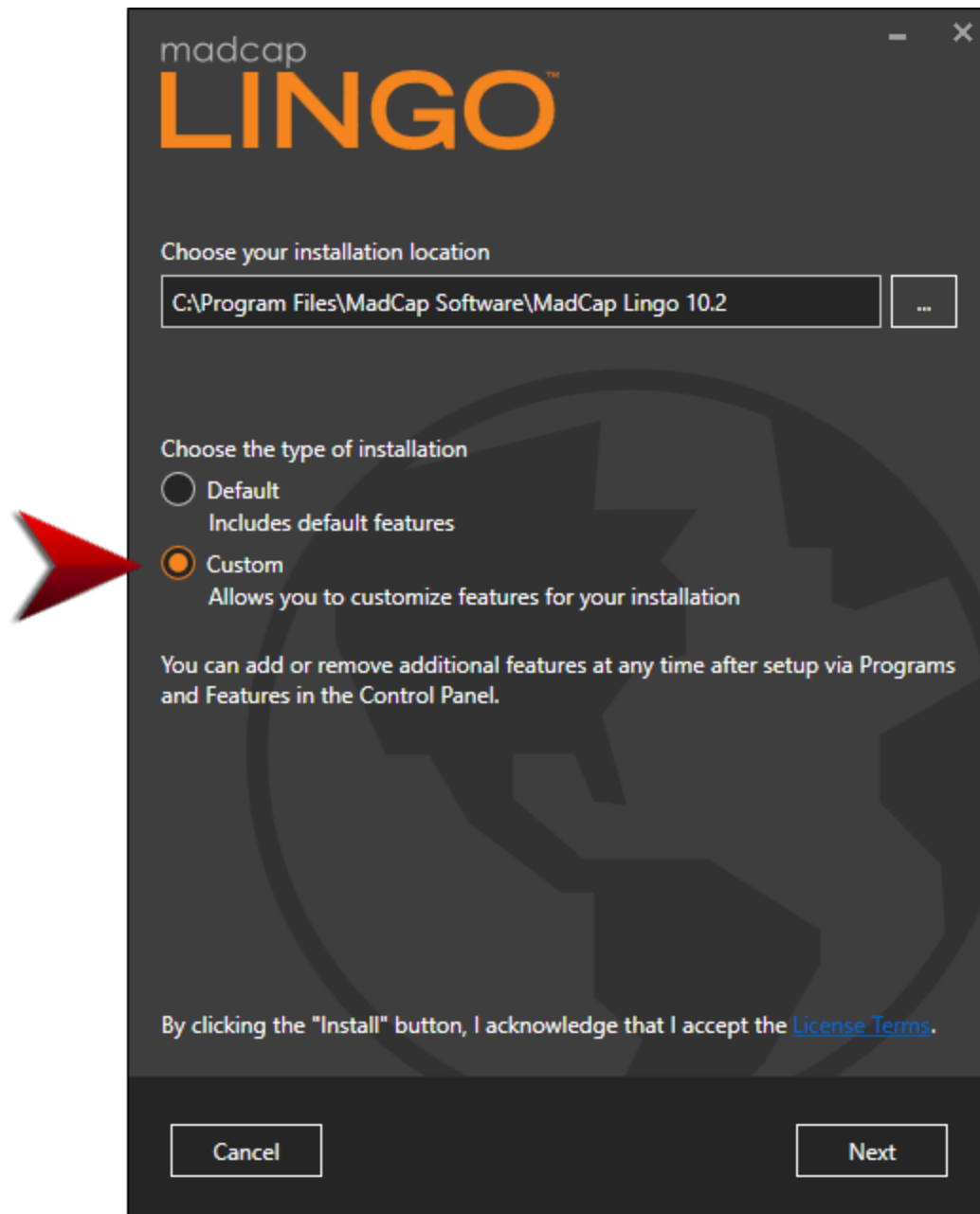
After you choose **Custom** and click **Next**, you can select the following option:

- **Local Documentation** If you do not select this option, only the external Help for Lingo will be available. This is beneficial because you are ensured of seeing the latest documentation, which is updated periodically on a server. Also, it reduces the overall size of the Lingo installation. However, if you want to include the Help system on your computer, select this option. You might choose to do this if you have limited access to the Internet or are behind a firewall.

Find and Replace Enhancements

A couple of enhancements have been made to the find and replace features in Lingo.

FIRST RESULT PER FILE

When you use the Find and Replace in Files window pane, there is a new option for customizing results. This option is called "First result per file."
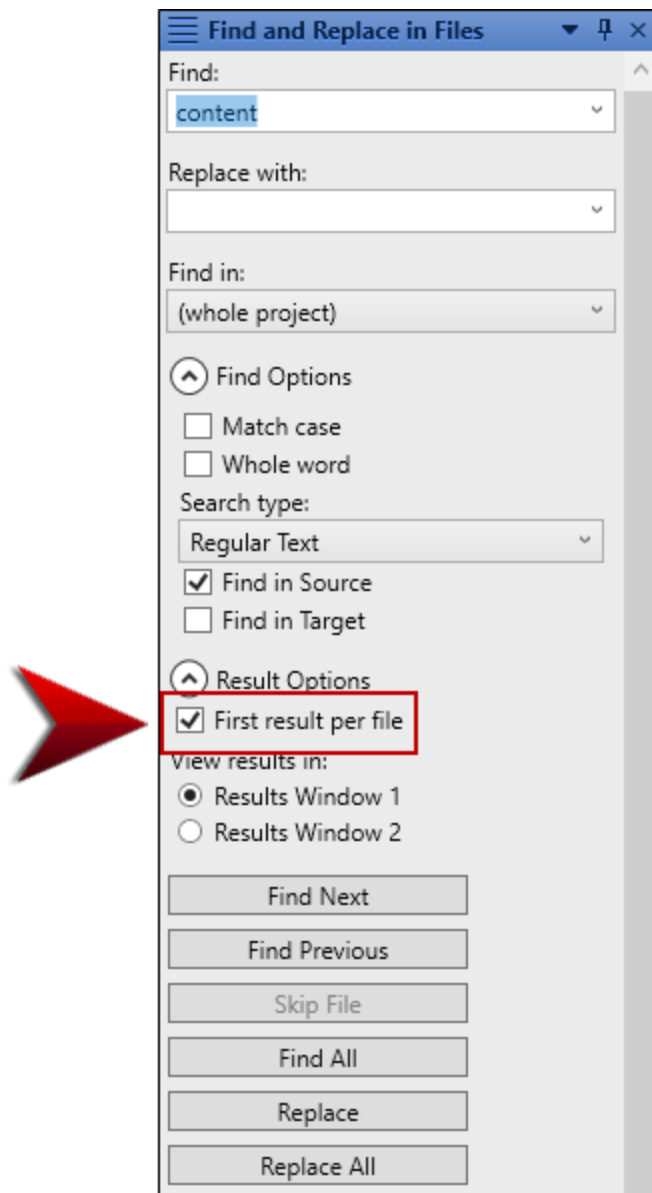
By selecting this option, you can limit the results shown so that a file displays only once, even if the term is found multiple times in it.

> ☆ **EXAMPLE**
>
> Let's say you search in all your source segments for the word "content," with the "First result per file" option disabled. In several files, the word is found multiple times, so a unique row is displayed for each occurrence. For example, the word is found six times in the file Appendix.htm, so there are six rows in the results for that file.

☆ Next, you enable the "First result per file" option and try again. This time, there are far fewer rows in the Find Results window pane, including just one for the Appendix.htm file.

After opening one of these files that contains the word multiple times, you can use the Quick Find widget to locate each concurrence of the word in the file.

QUICK FIND AND REPLACE WIDGET DROP-DOWN

The Quick Find and Replace widgets have been updated to offer a drop-down arrow next to the Filter button. Clicking this drop-down arrow gives you specific filtering options. This widget displays within a file.
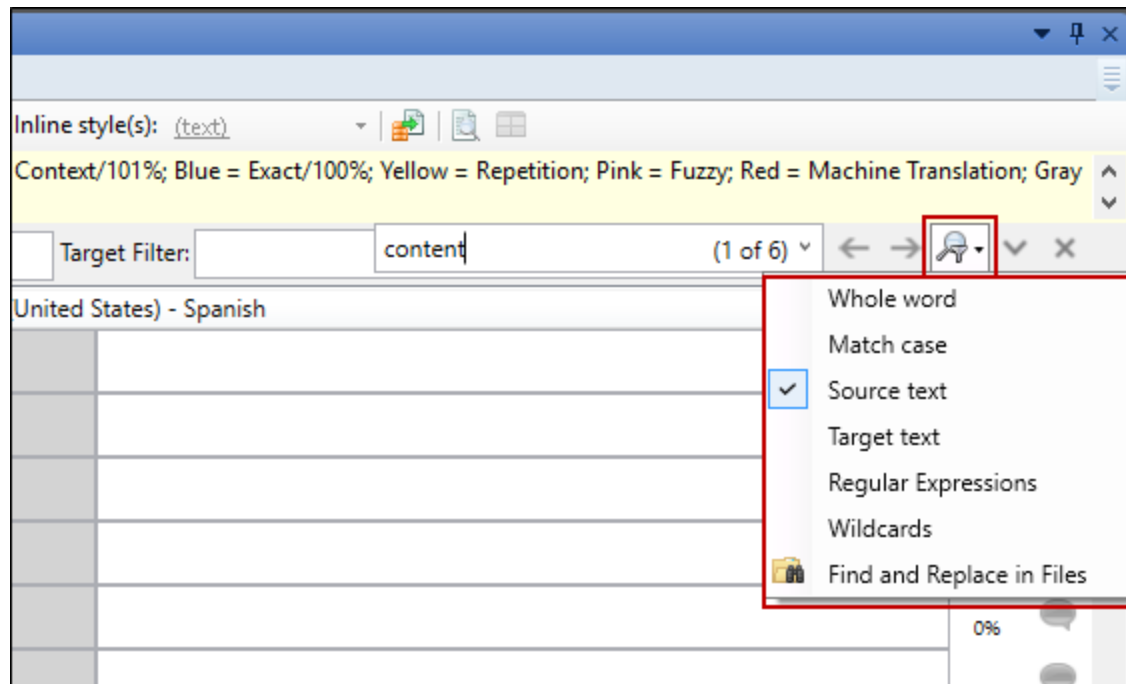
Microsoft TFS 2017/2018 Support

If you are using Microsoft Team Foundation Server (TFS) for source control, be aware that Lingo now supports TFS 2017/2018.

APPENDIX

PDFs

The following PDFs are available for download from the online Help.

Alignment Guide

Getting Started Guide

Key Features Guide

Shortcuts Guide

Source Control Guide: Git

Source Control Guide: Perforce

Source Control Guide: Subversion

Source Control Guide: Team Foundation Server

Termbases Guide

Touring the Workspace Guide

Translation Guide

What's New Guide