Localization

Agiloft supports international language localization. To fully translate the interface requires interaction with three files:

- An EUI Templates record, which applies to the end-user interface
- The user translation file, which is knowledgebase-specific text, which also supports machine translation using Google Translate Integration
- The system properties file, which applies to system properties for all systems running on the server

All system text is held in a single file; the system-wide localization controls the general program text, such as the text within setup wizards, error messages, and so on. A second file controls the terms specific to each knowledgebase, such as the names of tables, field labels, and header text. The end-user interface text is stored separately in an EUI Templates record.

> ⚠ Because the system properties file applies to all KBs on the server, you must have administrative access to the server to upload dictionary files. For hosted or shared servers, contact Support for more information about providing or collaborating on a system properties file for your target language.

Google Translate Integration

If you set up Google Translate Integration, you can generate machine translations for the user translation file. You can review machine translations using the review workflows described in the Google Translate Integration page, or you can download the user translation file for the target language and include previously translated terms. Generally, the in-system review workflows are more user-friendly than working with the translation files, unless you want the same person to process multiple files at the same time.

If you don't want to use Google Translate, simply download and process the user translation file using the same steps as the system properties file.

Download Translation Files

To access the EUI localization:

1. Expand the System table.
2. Open the EUI Templates table.
3. Open the `translation.properties` record.
4. Copy the text in the Body field.
5. Paste the text into a plain-text editor, such as Notepad++.
6. Save the file as `translation_xx.properties`, where `xx` is the java locale for the selected language. For example, a German translation file would be saved as `translation_de.properties`.

To download the localization files:

1. Click the **Setup** gear in the top-right corner and click Localization.
2. On the Language tab, select the target language from the drop-down. Click Next.
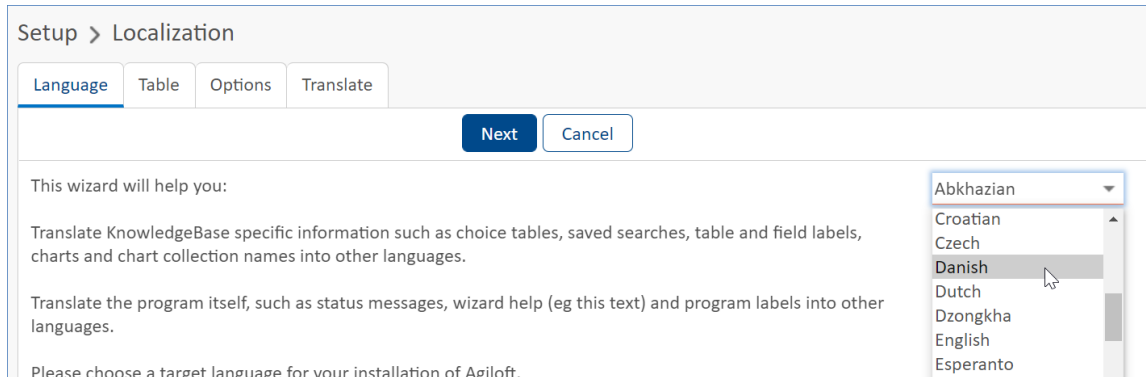
Selecting the target language

3. On the Table tab, select the Table Tree at the top to create a translation file for all the tables in the system. Table names appear in red when there is no available translation. To translate the text in a specific table, highlight the table name you wish to translate. Click Next**.**
4. On the Options tab, choose whether to exclude previously translated text by toggling the 'Hide fields that have already been translated' checkbox. Click Next**.**
5. On the Translate tab:
   - If you want to use Google Translate for the language you selected, select Machine Translate now and download the `language.txt` file. The system runs the content through Google Translate and applies the new machine translations to the file before you download it, so the downloaded file contains a full machine-translated dictionary for the language, which you can manually review and edit before uploading it to the system. You must set up Google Translate integration for this option to appear. If you have set up Google Translate integration and the option is not available, go back to your Google Translate configuration and make sure the language is selected in the list.
   - If you don't want to use Google Translate, simply download the `language.txt` file.
6. If you plan to localize the system text for a KB on your own server, download the `GUI2_xx.properties` file as well, where `xx` is the java locale for the selected language. This file contains the system text, while `language.txt` contains the customizable knowledgebase text, such as table and field names.

Localize the EUI File

Open the file where you saved the text from the EUI template (for our example, this is the file stylized as `translation_xx.properties`). The text contains 'key=value' pairs, where the key refers to a specific item like a table name, and the value is either a setting or a localized field name value. The body text indicates what the value means in each section. You can show or hide tables, define specific home and search pages, or define localized name values in any combination, but make sure those settings are appropriate to be tied specifically to the selected locale.

At a minimum, to provide a translated interface, translate the items under the heading `#These variables provide the text on the home pages.`

When you've translated each value and made any other desired changes, you can apply your translation to the system:

1. Go to the EUI Templates table.
2. Select the checkbox next to the `translation.properties` record.
3. Select **Actions** > **Copy**.
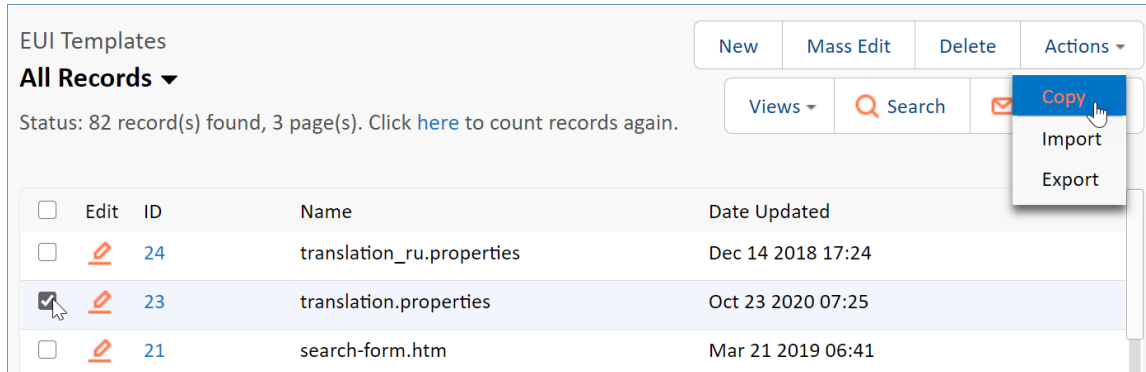
4. When prompted, name the record `translation_xx.properties`, where `xx` is the java locale for the selected language. For example, for Danish, add `_da` after `translation`.

5. Click Finish.
6. Open the new record.
7. In the Description field, replace English with the appropriate language.
8. In the Body field, delete all the text. Then, paste the text from your completed translation file into the field.
9. Click Save.

The EUI localization will be stored in your knowledgebase only, and does not interfere with other knowledgebases on the same server. It will remain in the knowledgebase after reimport and when copying the knowledgebase.

Localize the User Translation File

There are two ways to localize the user translation file in a knowledgebase: you can manually translate each named object, or use the localization file containing *all* custom text. The text is stored in a file.

> ⊘ With Google Translate Integration, you can automatically generate Google Translate machine translations for all text in the localization file, and simply review the machine translations for accuracy. When adding a new language to a system, this can be more time- and cost-effective.

Translate All Custom Text

To translate all knowledgebase text at once, you can download a single localization file, translate it, and upload it back to the knowledgebase. If you use Google Translate integration, you can automatically translate the file at the point of download, so you only need to review the file and then upload it.

The localization file is a property file. It contains 'key=value' pairs, where the key refers to a specific item like a field name, and the value is the localized field name value. The localization file is stored in UTF-8 encoding and can be edited in any program that supports that encoding.

1. Download the translation file for the target language, using the Enable Machine Translation option if desired.
2. Edit the `language.txt` file and translate or review each value. Before you begin working with the file, review these Localization File Tips.
3. Upload the translated file to the knowledgebase.

The localization file is stored in your knowledgebase only, and does not interfere with other knowledgebases on the same server. This file is retained even if the knowledgebase is re-imported or copied.

> ⚠ Remember that knowledgebase text does not include all the text you see in the system. The text on buttons and other interface items is controlled by the System Properties File.

Uploading the Translation File

Make sure the translations in the file are reviewed and approved before uploading the file. Uploading a file automatically marks all the translated content as Translation Approved for the Google Translate review tool.

After the file is translated, follow these steps to upload it to a knowledgebase:

1. Go to **Setup > Localization**.
2. On the Language tab, select the language you wish to update from the drop-down.
3. Navigate to the Translate tab.
4. Click Browse... to find and select your file, then click Upload.
5. Save your changes.

Translate as You Go

You can translate an individual item using setup menus. This workflow is designed for admins working in the language who spot an untranslated item and want to provide a translation on the spot.

✅ Google Translate Integration provides dedicated workflow options for translating new tables, fields, and other system entities.

To translate an individual item:

1. Select the target language from the Language menu drop-down in the upper right. For example, if you want to provide a French translation for an English item, choose French.
2. Navigate to the setup wizard for the item you want to translate.
3. Enter the translated values for the label.
4. Save your changes.

You can use this method to relabel a table, field, label, choice field item, admin description, etc. The changes you make are stored and kept in the appropriate language file for the knowledgebase.

Localize the System Properties File

System properties are server-wide and stored in the GUI2.properties file. You can download the file through the Setup menu of any knowledgebase.

See Localization File Tips for help, or contact support.

⊘ If you upload a customized file directly into your system, it will never be overwritten with future text changes or fixes from Agiloft. Please send your localized GUI2.properties file to Support after translation so it can be included in a future release build. Submit a ticket through the Support Portal with the localized GUI2. properties file attached after the translation has been completed.

Notes

- Localization does not include the ability to translate record data.
- Changing knowledgebase-specific text can be done by users with system admin privileges in the knowledgebase, and is KB-specific.
- Changing the system properties file requires server-level admin access to the installation directory. We recommend contacting Support to assist you.
- If you want all new knowledgebases created on your server to include the localization file, translate the default Demo Knowledgebase. New knowledgebases are created by based on the Demo template.