Translation Assistance Tool

for v7.0 or later systems

⚠️ Important changes are listed in Document Revision History at the end of this document.

Contents

What is the Translation Assistance Tool?

The Translation Assistance Tool allows you to:

- *Generate key terms* (page 1) to be translated (creates .native files)
- *Export the list of key terms* (page 2) to send to a translation house (creates .translate files)
- *Import a translated file of key terms* (page 3) into the system files (overwrites the .native files in Bundled Resources)
- *Run a report* that identifies key terms that have not been generated, translated, or used (creates .csv files)
- *Revert the generation of key terms* (reverts a .logic file to its original untranslated state)

Prerequisites

You must have:

- a v7.0 or later system with all needed language packs installed
- a text editor that supports UTF-8 encoding, such as NotePad++
- the Improved Edit Order tool

⚠️ **CAUTION**   If the control program has been translated using Translation Assistance Tool, use Improved Edit Order instead of Edit Order for all future reordering of microblocks.

Running the Translation Assistance Tool

Step 1: Generate key terms

Generate key terms to:

- create a .native file containing a list of all translatable key terms in each language selected.
- display an "@" in front of each translatable key term in the workspace.

NOTES:

- The key terms are linked to content for specific translatable property fields — State Text, On/Off, or Point Name.
- If translating Property Text, only those microblocks with Show Property Page Text selected will generate a key.

1   Click **Translation** > **Generate Keys**.

2   If you are going to upload the symbol or equipment into a SAL database for use in EquipmentBuilder, check the **Include sal fields** checkbox.

  NOTE   This generates the description, alarm text, and return text key terms for translation.

3   Select the source file(s) to be translated. Source files can be .equipment or .logicsymbol:

  ○   **Currently loaded** - the file currently displayed in the workspace

  ○   **All in directory** - browse to the directory where the logic files reside

4 Select your base language from the droplist.

NOTE   Base language is the language that the programmer used when programming the logic and its properties; to translate the key terms, the base language must be ISO 8859-1.

5 Select the languages from the list that you want translated.

6 Click **Generate**.

An additional file is created for each language selected above. Each language file has a 2-letter extension appended to the file name to indicate the language. All language files contain a list of key terms to be translated.

7 If generating keys only for the currently loaded file, click **File** > **Save**.

NOTES:

- Additional languages can be translated on the **Generate Keys** page by selecting the new languages and **Generate**.

- If more than one key begins with the same first four words, such as alarm text, the first key will be truncated to the first four words. The second instance is appended with "2" as in the `Filter_Status2` example above. Each subsequent key will be appended with a digit increased by 1, such as `Filter_Status3, Filter_Status4`, and so on.

- If more than one microblock contains identical text for a particular translatable property field, the same key will be used for all of them. Translations for that key will apply to all microblocks containing that key. For example, you may have multiple microblocks with the @off key with identical translations.

Step 2: Export to .translate file

Key files can be used to create a .translate file for each language that you want translated. The file can be viewed and edited using your text editor. The .translate file can be given to a translation house to provide equivalent translations for your key terms.

To export the files for translation

1 Click **Translation** > **Export**.

2 Select the source file(s) to be exported:

- ○ Currently loaded file (displayed in the workspace)
- ○ All in directory (browse to the directory)

3 Select directory to export the files to.

4 Click **Open**, then click **Export**.

To view and translate the key terms

1 Browse to the location you selected above in step 2.

The folder contains a .translate file for each language containing the key terms in the base language.

EXAMPLE: If you selected French, German, and Italian, and your base language is English, you will see a file for each .equipment or .logicsymbol file ending in _en.translate, _fr.translate, _de.translate, and _it.translate.

2 Use your text editor to view the contents of the files.

EXAMPLE: If English was selected as your base language, this is an example of two of the items in the English file:

```
@Filter_Status=Filter_Status
@Filter_Status2=Filter_Status2
```

3 Translate, or if desired, send the `.translate` files to a translation house.

To translate, one must replace the right side of each key term with its language's equivalent. Do not translate any formatting characters or text between quotation marks since that is reserved text. EXAMPLE: The French translation house should return these changes:

```
@Filter_Status=Filtre_Statut
@Filter_Status2=Filtre_Statut2
```

For more information on translating text, see the *Appendix* (page 5).

Step 3: Import .translate files

After translating or receiving the .translate files back from the translation house, you can import the changes into the system.

1 Click **Translation** > **Import** to convert the .translate files to .native files used to translate the Property Page.

2 Browse to the directory of the .translate file(s).

3 Select the control program(s) to be imported to:
- ○ Currently loaded file (displayed in the workspace)
- ○ All in directory (browse to the directory where the logic files reside)

4 Click **Import**.

Step 4: Run a report

After the translation files have been imported, you can run a report on each piece of equipment that identifies any key terms that have not been generated, translated, or used. Each equipment will have its own .csv file.

To run the report

1 Click **Translation** > **Report**.

2 Select the control program(s) to include in the report:
- ○ Currently loaded file (displayed in the workspace)
- ○ All in directory (browse to the directory where the logic files reside)

3 Browse to location you would like the file saved to.

4 Select the languages from the list to be included in the report.

NOTE   The base language is grayed out and will automatically be included in the report.

5 Click **Run**.

To view the report

1 Using Excel©, open the .csv file created above.

2 Review the list of key terms evaluated in the first column under **Type**. See table below.

3 If any items on the list still need to be translated, re-generate keys, re-export, re-import, and re-run the report.

4 Address the errors or send the report and .translate files to the translation house to address the translation errors.

5 After receiving the revised .translate files back, re-import the .translate file, re-run the report, and review the list of key terms again.

6 Upload into Salamander if you intend for the equipment to be used in EquipmentBuilder and you checked **Include sal files** when you generated the key terms (see *Step 1: Generate key terms* (page 1)).

7 Repeat this process until the translations are complete.

In this example, English is the base language and additional languages are displayed in columns to the right. 3 of the 4 possible error types are shown. See table below for more information.

	Error Type	Description
a	**Missing Translations**	The specific microblock is missing one or more translations.
b	**New Keys**	Nothing was translated for this key.
c	**Unused Keys**	Keys that had previously been used, but aren't currently in the .equipment file; the file name may have been changed. NOTE An example of this would be deleted microblocks; having unused keys does not affect your translations.
	Ungenerated Values	New key terms have been added since the last key generation. NOTE Update the file by generating, exporting and importing your key terms again. Then run this report again to verify the file is up-to-date.

Reverting files back to original state of .native

By reverting, you can start the translation process all over again. For example, you may have selected the wrong base language or selected **Include sal fields** by mistake.

You can revert all translatable fields back to their values in the base language, remove the generated keys and revert the logic back to its original untranslated state.

1 Click **Translation** > **Revert**.
2 Select the source file(s) to be reverted:
   ○ Currently loaded file (displayed in the workspace)
   ○ All in directory (browse to the directory where logic files reside)
3 If you want to revert to the original .logicsymbol or .equipment file, also click **Control Programs** > **Bundled Resources** and remove the .native files.
4 Click **File** > **Save**.

Using the Translation Editor

You can view or edit how key terms appear in different languages using the Translation Editor.
1 In the workspace, select a microblock.
2 Click **Translation** > **Translation Editor**.
3 Under **Working On**, view the microblock's name and base language.
4 From the **Language** droplist, select the language that you want to display or edit.
5 Edit the text as desired.
6 Click **Close**.
7 Click **File** > **Save**.

Appendix A - Translation guidelines

TIP   Do not translate Reserved Text or property page formatting characters.

Property Page Formatting Characters

A pair of **$** characters indicates Reserved Text and is not be translated. If you translate it, the property's value will not display.

In the examples below, the properties and their formatting characters are in blue and should never be translated. You must **only** translate the black text and characters.

Reserved Text	$...$	If method is "Manual Clear," $Present_Value$ runtimers now.

Formatting characters, such as those used for bolding or carriage returns, should never be translated.

In the examples below, the formatting characters are in green and should never be translated. You must **only** translate the black text and characters.

Carriage Return	^\	^\ Setpoint above is based on effective setpoint from "Setpoint Section".
Omit Carriage Return	^$	If method is "When Manager Off," ^$ Then new staging order resets only after manager is off.
Bold Font	!{...!}	!{ Alarm(s): !}
Change Default Formatting	^##	^## + $relinquish_default$ $units:editable="false"$.

More examples

Only translate the black text in the following examples.

!{--- Defined Time for Runtimers Clear:!} All automatic clear methods (1,2 & 3 above) will occur ^\ at $Present_Value:control="minsec"$ (24hr format).

^##--- Limit current running chiller(s) power to: $relinquish_default$ $units:editable="false"$ whenever a new chiller stages on.

Document revision history

Important changes to this document are listed below. Minor changes such as typographical or formatting errors are not listed.

Date	Topic	Change description	Code*
12/21/20	Requirements Cover	Updated version support	A-D
9/22/20	Cover	Updated for v7.0 or later Removed Installing a patch	X-LO
9/24/18	What is...	Removed the term "add-on"	X-AE-LO-E
8/28/18	Installing a patch	New topic	A-RR-LO

* For internal use only