



**CCL Localizer  
Reference Manual**

## Introduction

CCL Localizer is a simple translations editor specifically tailored for editing string tables used with localization of applications based on the CCL Cross-platform Framework like PreSonus Studio One. It primarily supports the workflow and file formats of GNU gettext, as well as simple text files.

For more information on GNU gettext please refer to <http://www.gnu.org/software/gettext/>

This manual is targeted to translators which are already familiar with the basic concepts of software localization. PreSonus provides all necessary .po/.pot files for its products which can be edited either with CCL Localizer or any other professional localization tool if advanced features like Translation Memory are required.

Besides translation editing, CCL Localizer is able to create language packs, which contain all resources for a language in a single file. Support for a new language can be tested simply by putting the .langpack file in the "languages" subfolder of the program, without having to recompile or reinstall it.

## The CCL Localizer Window

After launching CCL Localizer its main window appears. The main parts are Files List, Context List, String List and the Edit/Search area.

### The Files List

Choose File/Open from the menu bar to open a folder with .po or .txt files. The files list will be populated with all files found in this folder and show a statistics of how many items per file are already translated. When restarting CCL Localizer it always tries to restore the last selected folder.

The icon next to the file name indicates the file state:

- green = all strings translated
- yellow = some strings not translated
- asterisk = file has been modified but not saved

When files have been modified, please note that choosing File/Save from the menu bar will save all files at once.

The files list can be used to filter the items displayed in the string list on a per-file basis. Double-click on a filename to show its items exclusively or use the check marks either with single-click or Alt-click to set the desired filters. The filter can be reset either with the View/Reset Filter command or by right-clicking and choosing "Reset Filter" from the contextual menu.

## The Context List

The context list shows the available context identifiers depending on which files are currently loaded and activated in the files list. A context string defines where a specific translation is used in the program and it also helps with disambiguation in the target language when the same English original needs different translations.

The context list can also be used as a filter in the same way as the file list to narrow down the items displayed in the string list.

## The String List

The string list shows all items for translation in a flat view, depending on the filters defined in the files and context lists. It can be filtered and sorted further in various ways. Translations can be edited directly in the string list via double-click and you can navigate between items with the cursor and Tab keys. The focused item is always displayed in the edit area.

The most interesting view when adapting to changes made in the program is to show untranslated and orphaned items. Orphaned items occur when an English original which has been translated already is changed or removed in the program. They appear in red text color in the string list. After review, orphaned items can be removed permanently from all files by using the Edit/Cleanup command.

When an item has been modified but the file was not saved yet, the text color of this item changes to blue. Additionally, if the current filter would cause the item to go out of view immediately after editing, it is kept visible until the file has been saved.

To copy translated text between items, simply drag-and-drop them onto each other.

## The Edit Area

The edit area shows a multiline view of the item currently focused in the string list as well as navigation arrows to copy the English original to the translation and to focus the next/previous item. These commands are also available in the main Edit menu.

On the right side a list shows occurrences of the same English original in all available contexts. This helps to keep a good overview to create a more consistent translation. Similar to the string list, items can be copied via drag-and-drop. A double-click will focus the item in the main string list.

The edit area can be flipped to a search field with several options. The items matching the search terms will be listed below. It features the usual drag-and-drop and double click behavior as the other list views.

When editing translations the English original often contains placeholders like "%(1)", "\$APPNAME", "@cmd.title[Edit|Copy]", "@value[]" or escaped character like "\n", etc. This placeholders need to be preserved in their correct form in the translated text. The order of arguments like "%(1)" can be changed to accommodate to the grammar rules of the target language.

## The File Menu Functions

Open	Open a folder with .po or .txt files
Save	Save all modified files
Revert	Revert all files to their state on disk
Close	Close all files
Update	Update loaded .po files from templates. A folder containing the .pot template files needs to be selected. Files will be matched by name, i.e. xyz.po will be updated with xyz.pot.
Compile All	Create binary .mo files for all loaded .po files in the same folder
Import	Import translations from file. All currently loaded items with a matching context and English original will be updated with the translation loaded from the chosen file.
Export	Export all items currently visible in the string list to a new .po file. The import/export functions can be used to send a subset of the whole translations project to a translator when e.g. only items that changed between program versions need to be worked on.
Quit	Quit the CCL Localizer

Please note that when editing .txt files with CCL Localizer, the file name will be displayed in the original column and the file content in the translation column. The content of the file will be replaced during the translation process. If the .txt file does not contain a BOM (Byte Order Mark) UTF-8 encoding will be assumed.

## The Edit Menu Functions

Undo / Redo	Undo/redo last translation
Copy / Paste	Copy/paste selected translation
Set To Original	Set translation of focused item to the English original
Strip Context	<p>The CCL framework prepends the context identifier to the English original in square brackets when saving files in PO format*, which must not be repeated in the translated text. This is transparent to the translator when working with CCL Localizer, but not with other localization tools. The Strip Context functions removes any accidentally repeated context identifier from the translation.</p> <p>* Future versions might use the <i>msgctxt</i> attribute defined in PO format.</p>
Cleanup	Remove orphaned items from all files
Pseudo-Localization	This set of functions allows to create pseudo translations by decorating the English original or the existing translations with special Unicode characters readable to English speakers for testing purpose.

## **The View Menu Functions**

Filter by Context	Set the context filter to all context identifiers found in the currently selected items of the string list.
Filter by File	Set the files filter to all files found in the currently selected items of the string list.
Reset Filter	Reset all filters completely
Previous / Next Item	Navigate between items in string list

## **The Tools Menu**

Build Language Pack	Create a language pack. For this function to work, you need the required build files provided by PreSonus. All currently open files will be compiled first, because most often the language pack will be build for the translations which are currently being worked on.
---------------------	--

## **The Help Menu**

Contents	Brings up this reference manual
Language	Select the GUI language for CCL Localizer
About	Show version and copyright information of CCL Localizer

## Appendix A – Translations Folder Structure

PreSonus provides localization files for its products in the following folder structure.

<translations>	Root folder
<icons>	Icons in PNG format for each language
<locales>	XML files with locale information per language. It defines day and month names, as well as the date/time format.
<projects>	This folder contains the build files for language packs sorted by product. If you do not have this folder, you can still translate string resources, but you will not be able to create language packs on your system.
<resources>	This folder contains additional resource files which need to be translated besides the .po files.
<strings>	All .po files are located here within subfolders per language.
<tables>	String table definitions used by the language pack build process (optional for translation editing only)
<templates>	Language-independent template files in .pot format.

## Frequently Asked Questions

*Q: I have updated the .langpack file in the program "languages" folder, but some of my new translations do not show up in the program.*

A: The program might cache some translated strings in its settings files. Please increment the "LanguagePack:Revision" number in the metainfo.xml file of your language pack. Alternatively, you can open the settings folder of the program (Help/Open Settings Folder in PreSonus Studio One) and delete "x86/Plugins-xx.settings" and "x64/Plugins-xx.settings" (xx is the language code).

## **Appendix B – Description of "Navigation" commands for translators**

Important concepts for understanding the navigation commands are the focus item (the item at the current navigation position) and the selection. In an Explorer/Finder-like file browser view, selected items appear with a highlighted background color, while the focus item is marked with a dotted rectangle.

A navigation command starts with a direction, optionally followed by options.

These commands usually move the focus item in the given direction. Possible directions are Left, Right, Up, Down, Page up, Page Down, Start, End.

The options determine how the selection changes:

- (none): selects only the new focus item

- Extend: selects a range of items between the old and the new focus item

- Add: all previously selected items stay selected

- Skip: selection is not changed at all, only the focus item moves

With these commands you can make a non-contiguous selection without using the mouse, e.g. select a range of items (extend), skip over some items and add another range (extend add).