

Database based configuration

Translate5 configuration

translate5 has a lot of administrative configuration options.

Nearly all of the configuration options can be set in the preferences section of translate5, if you are logged in with admin rights:

Configuration option	Value	Code ID
Moses default match rate Default fuzzy match value for translations done by Moses. Used in the analysis and in the fuzzy match panel. If ModelFront is not used for risk prediction of MT.	70	runtimeOptions: LanguageResources.mo
Moses server URL(s) Zero, one or more URLs, where a Moses server is accessible as a language resource. Example: http://www.translate5.net:8124/RPC2	http://localhost:8124/RPC2	runtimeOptions: LanguageResources.mo
Only TermCollections assigned: Hide fuzzy match panel If set to active and only a TermCollection and no MT or TM language resource is assigned to the task, the fuzzy match panel will not be shown in translate5s editor. Overwrite levels: Client, Import, Task	Activated	runtimeOptions.editor: LanguageResources.mo
OpenMT2 instance pre-fix When using one OpenMT2 instance for multiple translate5 instances, a unique prefix for each translate5 instance must be configured to avoid filename collisions of the Memories on the OpenMT2 server.	1	runtimeOptions: LanguageResources.op
OpenMT2 server URL(s) Zero, one or more URLs, where an OpenMT2 server is accessible as a language resource. Example: http://openmt2.translate5.net:1984/otmemoryservice/	http://localhost:1984/otmemoryservice/	runtimeOptions: LanguageResources.op
OpenMT2: Show all 100% matches If this is set to disabled, for 100%-Matches that differ in the target, the target of the match with the highest match rate is shown. If the match rate is the same, the match with the newest change date is shown. If set to active, all 100%-Matches that differ in the target are shown. Overwrite levels: Client, Import, Task	Disabled	runtimeOptions: LanguageResources.op
Preload fuzzy matches in advance For how many segments starting from the current one in advance fuzzy matches are pre-loaded, so that they are immediately available for the translator?	1	runtimeOptions: LanguageResources.pre
Resource usage log lifetime How many days the resource usage logs will be kept in the database.		runtimeOptions: LanguageResources.us

Alternatively it is recommended for system administrators to change translate5 config options on the command line - as explained here: [./translate5.sh config](#)

since server administration is mostly done on CLI too, and some rare config values are only changeable via command line:

```
> tlauria@tenforward ~/www/translate5-master P develop 6 > ./translate5.sh config cronip
Change Translate5 configuration.
=====
HostName: translate5.localdev
AppRoot: /home/tlauria/www/translate5-master
Version: development

Configuration found:
-----
name: runtimeOptions.cronIP
value: 127.0.0.1
default: 127.0.0.1
category: cron
  GUI: System setup: General / IP address allowed for cron calls
defaults:
  type: string
  level: instance - 2
description: It is recommended to call translate5s cron job mechanism every 15 min. These calls are only allowed to originate from the IP address configured here.
comment:
```

Some configuration explained in detail

Language resource configuration

The following language resources can be configured for use in translate5:

- DeepL (as Plugin, only available for companies with a Support- and Development contract)
- Google
- Lucy
- Moses default
- MS Translator
- SDL LanguageCloud
- Globalese (not as part of Language Resources, but with a special integration)
- For SDL GroupShare please ask MittagQI

To integrate a [language resources](#), search in the translate5 preferences panel for "languageresources" - see the first screenshot here. Or enter in the commandline:

```
./translate5.sh config languageresources
```

This will give you the configuration values for all existing language resource integrations and some general configuration parameters for language resources.

How to configure each value is explained on each configuration value.

Help Page configuration

It is possible to show a help button in translate5. The button opens a pop up window with help content.

The displayed content must be provided as HTML-files to translate5 by the company using translate5.

At the moment no default help page content is provided by the installation.

The HTML-files that are provided as help content are loaded by AJAX and displayed in Popup Window in translate5.

When the configuration loaderUrl for the help files (details see below) is empty, the help buttons are not shown at all.

How to configure the help button

For each tabular view in translate5 (task overview, user administration, clients, editor and language resource) the corresponding help page can be configured separately.

By default the window is automatically shown, if the help loaderUrl for the corresponding tabular view is not empty.

The loaderUrl for each tabular view can be defined in the following configurations:

1. runtimeOptions.frontend.helpWindow.customeroverview.loaderUrl
2. runtimeOptions.frontend.helpWindow.taskoverview.loaderUrl
3. runtimeOptions.frontend.helpWindow.useroverview.loaderUrl
4. runtimeOptions.frontend.helpWindow.editor.loaderUrl
5. runtimeOptions.frontend.helpWindow.languageresource.loaderUrl

The default configuration in translate5 looks like the following:

Table view	configuration name	value
Customers	runtimeOptions.frontend.defaultState.helpWindow.customeroverview	{"doNotShowAgain":false}
Task overview	runtimeOptions.frontend.defaultState.helpWindow.taskoverview	{"doNotShowAgain":false}
User administration	runtimeOptions.frontend.defaultState.helpWindow.useroverview	{"doNotShowAgain":false}
Editor	runtimeOptions.frontend.defaultState.helpWindow.editor	{"doNotShowAgain":false}
Language resources	runtimeOptions.frontend.defaultState.helpWindow.languageresource	{"doNotShowAgain":false}

Define "automatic show"-behaviour of the help page with the "doNotShowAgain" config option

The value "doNotShowAgain" is by default set to false for all tabular views. This means:

- as soon as a loaderUrl config for the tabular view is defined, the help page will automatically be shown once, as soon as the user enters the relevant section.
- After each new login it will be again shown once.
- For each help page the user can click a checkbox, that he for himself does not want to see this help page again automatically

If the value "doNotShowAgain" is set to "true" for a tabular view in Zf_configuration, the corresponding help page of that tabular view will never pop up automatically for any user.

loaderUrl configuration definition

For all tabular views no default loaderUrl config is defined. Thus no help button will be displayed.

If you want to display a help button for a tabular view, enter a value.

If you enter complete loaderUrl with protocol and domain like <https://www.youtube.com/yourvideo>, this loaderUrl will be shown as content of your help page.

If you enter a relative value, this needs to start with /help/ and translate5 will look in the folder

/application/modules/default/views/scripts/help/

for a phtml-file that is rendered.

However you should not place your custom phtml file in this folder, because it will be overwritten/deleted with the next update.

Instead use the feature to [Implement a custom translate5 skin / look & feel](#) and put your custom phtml help page in the client-specific folder, where it will not be overwritten.

Help window configuration in editor

You can configure the help window and show content/documentation in 2 different tabs from 2 different sources.

- Videos
 - defined with ***runtimeOptions.frontend.helpWindow.editor.loaderUrl***
 - default `/help/{0}` . The default will show the translate5 editor video from the translate5 youtube channel
 - the content is loaded from editor.phtml located in `/application/modules/default/views/scripts/help/` overwritable with client specific functionality
 - is meant for displaying video content documentation
 - can be configured to load the content from remote url. The remote content will be loaded into an iframe
- PDF documentation
 - defined with ***runtimeOptions.frontend.helpWindow.editor.documentationUrl***
 - default `/help/editordocumentation/{0}` . The default will show the written pdf docu for the translate5 editor.
 - the content is loaded from editordocumentation.phtml located in `/application/modules/default/views/scripts/help/` overwritable with client specific functionality
 - is meant for showing written documentation.
 - If you provide placeholder (`{0}`) in the url ex: `http://translate5.local/editor?lang={0}` , the language locale of the current user will be set when the url is called.
 - can be configured to load the content from remote url. The remote content will be loaded into an iframe

Help page localization

The current locale of the authenticated user can also be passed to the loaderUrl via the optional placeholder "`{1}`". This fragment is replaced with the current users locale.

Also the whole loaderUrl can point to a completely different web page:

Example:

```
"http://www.example.com/translate5-help/?section={0}&language={1}"
```

Pass section by parameter

You can also make translate5 to pass the current tabular view name in translate5 as parameter in the help page loaderUrl via the optional parameter "`{0}`". The passed value is the last part of the Zf_configuration name for that view (see above table).

So for example for the task overview in the above table you see, that the Zf_configuration name is "runtimeOptions.frontend.defaultState.helpWindow.taskoverview" and thus the passed value for the task overview would be "taskoverview".

Activate commenting of locked segments in Zf_acl_rules

This following content describes, how access control list (ACL) values can be changed on database level, which is not recommended!

People without programming knowledge should NOT change anything in Zf_acl_rules, since it is possible to completely mess up your translate5 installation, if you do not know what you do.

Please only change these settings, if you really know, what you do.

Activate commenting of locked segments

To allow the user to comment locked segments, please execute the following sql-query on your translate5 database:

```
INSERT INTO `Zf_acl_rules` (`module`, `role`, `resource`, `right`) VALUES ('editor', 'basic', 'frontend', 'editorCommentsForLockedSegments');
```