LanguageResources: Instance

If the API changes in an incompatible manner this is listed explicitly in the "important release notes" section of each releases changelog.

Description

Enables the usage of third party TM and MT systems in translate5. To support a MT / TM ti must be implement as a service in translate5. This is quite easy and usually a matter of days. Please contact us, if you need help on this.

Moses MT, OpenTM2, Google, Lucy, MS Translator, SdlLanguageCloud, NEC-TM (as Plugin), PangeaMT (as Plugin), TildeMT (as Plugin, only available for companies with a Support- and Development contract) and DeepL (as Plugin, only available for companies with a Support- and Development contract) are supported so far. For SDL GroupShare support please ask MittagQI.

TM/MT systems (or in general language resource) can be associated to tasks. In such tasks the user gets automatically found matches to the segments he is editing.

Also a concordance search is provided.

Provides access to the data of the configured language resources (TM,MT and termcollections).

URL:	/editor/languageresourceinstance/[ID]
Available Methods:	GET / POST / PUT / DELETE
Specialities:	Non filebased langauge resource can only be added by the "admin" role. Users with the role "pm" can add filebased entries.

Special URLs

URL	Methods	Parameters	Description
/editor/languageresourceinstance/[ID]/query	GET	**segmentId (integer)** *the id of the segment to be queried*	returns a list with query results, format see below
/editor/languageresourceinstance/[ID]/search	GET / POST	**query (string)** *the string to be searched in the underlying service* **field (string - source or target)** *determines which field should be searched in a TM* **limit (integer)** *how many results should be returned*	returns a list with search results, format see below
/editor/languageresourceinstance/[ID]/download	GET	-	Returns the uploaded TM file - if any.

/editor/languageresourceinstance/[ID]/import	POST	**tmUpload** *TM file uploaded as multipart/form-data* **Additional parameters only available for termcollection import** **mergeTerms** *The merging of terms into an existing term collection is controlled by this boolean parameter.* *The merge process is complex and completely described here.* **deleteTermsLastTouchedOlderThan (date format Y-m-d)** *Deletes all terms in all listed termCollections, that have a updated date older than the date configured for "deleteTermsLastTouchedOlderThan". Every term in a termCollection has a creation date, a modification date and an updated dated. The creation date and the modification date are taken from the dates as they appear in the TBX file.* *The updated date is not shown in the GUI and is changed every time a user changes the term through touching it in the GUI or through an import.* **deleteTermsOlderThanCurrentImport (boolean)** *Deletes all terms in all listed termCollections, that have an "updated date" older than the start of the current import. To learn, what the "updated date" is, please read above possible values: true or false.* **deleteProposalsLastTouchedOlderThan (date format Y-m-d)** *Delete all proposals in a listed termCollections, that have a creation date older than this configs value.* **deleteProposalsOlderThanCurrentImport (boolean)** *Delete all proposals in a listed termCollections, that have a creation date older than the start of the current import.*	Imports the given TM file into the chosen Language Resource instance
/editor/languageresourceinstance/[ID]/tasks	GET / POST	GET **id (integer)** *ID of the task language resource association* **taskGuid (guid)** *TaskGuid of the associated task* **taskName (string)** *name of the associated task* **state (string)** *status of the associated task* **lockingUser (guid)** *user who is locking of the associated task - if any* **taskNr (string)** *task number of the associated task* **languageResourceId (integer)** *associated TM ID* POST **toReImport (array)** *an array of taskGuids. The chosen tasks segment data is reimported into the selected TM.*	Gives some status information about the tasks associated to a specific Language Resource. Triggers a reimport of the chosen tasks into the TM.

Search / Query result format

Name	Type	Info
attributes	string	Additional meta data of the returned match/search result, empty if not available.
created	date	Meta data, creation time stamp of the returned match/search result
creator	string	Meta data, creator of the returned match/search result
lastEdited	date	Meta data, last edited time stamp of the returned match/search result
lastEditor	string	Meta data, last editor of the returned match/search result
matchrate	integer	matchrate of a found match, 0 for search results.
source	string	The found source text.
state	string	"loaded" wenn all is OK, "servererror" when we got an error in the queried service.
target	string	The found target text.
languageResourceId	string	The queried Language Resource, just for reference.

Langauge-Resource Instance Resource Layout

Name	Type	Info
id	integer	GET only
entityVersion	integer	GET only, set by server
langResUuid	string	GET only, is ignored on PUT/POST, is an auto generated UUID set on language resource creation. Contains a unique identified for the language resource.
specificId	string	GET only, is ignored on PUT/POST, is an ID coming from service behind the language resource, if any. Default null.
name	string	Custom name of the match resource.
forced	boolean	DELETE only, If set to true, all task associations will be removed before delete
deleteLocally	boolean	DELETE only, if set to true, the resource will be only removed from the local translate5 instance. When set to false, the connector will try to remove this resource from remote if exist(ex: t5memory exist once as translate5 resource and once as memory on t5memory side)
sourceLang	integer	GET / POST, non editable after creation; numeric DB id of the language
targetLang	integer	GET / POST, non editable after creation; numeric DB id of the language
color	string	Custom color of the match resource in hexadecimal form, used in the GUI.
resourceId	string	POST / GET, non editable after creation
serviceType	string	POST / GET, only, set by setting a resourceId
serviceName	string	GET only, set by setting a resourceId
fileName	string	POST / GET, can not be changed
filebased	boolean	GET only, set by setting a resourceId
searchable	boolean	GET only, set by setting a resourceId
customerIds	array of integers	POST, PUT: contains an array with customerIds to be assigned to that language resource

customerUseAsDefaultIds	array of integers	POST, PUT: contains a subset of customerIds of the above customerIds. The here referenced customers use this language resource by default to read from.
customerWriteAsDefaultIds	array of integers	POST, PUT: contains a subset of customer ids of the above customerUseAsDefaultIds. The here referenced customers use this language resource by default to write into it.To be able to write by default the customer must be also set as read as default.
customerPivotAsDefaultIds	array of integers	POST, PUT: All customers listed here will use this language resource when assigned to a task for pivot pre-translation. All of the listed customers in customerPivotAsDefaultIds must be also contained in customerIds.

Configuration of the Language Resources

Config name	Values	Default	Description
runtimeOptions.LanguageResources.fileExtension	map		Available file types by extension per engine type. The engine type is defined by source rcf5646,target rcf5646. ex: "en-ge,en-us"
runtimeOptions.LanguageResources.preloadedTranslationSegments	integer	3	Number of segments for which matches are preloaded (cached) in the front-end.
runtimeOptions.LanguageResources.moses.server	json array	[ ]	Moses MT related: List of available Moses Server, example: ["http://www.translate5.net:8124/RPC2"]
runtimeOptions.LanguageResources.moses.matchrate	integer, 0 - 100	70	Moses MT related: Moses MT penalty value, used as default matchrate since in MT no matchrate is available.