ZIP import package format

Visit this page for example ZIP import packages.

Please note: translate5 project strives to support the TIPP format as import and export format. Any kind of support for this development is welcomed.

File structure of an import package

Type	Name	Usage	Only available with listed plugin	Mandatory
Directory	workfiles (proofRead is deprecated)	Contains a files and/or a folder tree with the files, that should be imported into the segment table of translate5. The folder tree inside of workfiles will be mirrored as folder tree inside of translate5.		yes
Directory	relais	Contains a structure that mirrors the structure in workfiles. This means, all directory and filenames must be identical with the names in the workfiles directory. All contained files are imported as relais/pivot content for the files in workfiles. Files existing in "workfiles" but missing in "relais" are skipped.		no
Directory	referenceFiles	Reference files that are attached to a task for download. Additional information for the user working on the task.		no
Directory	visual (visualReview is deprecated)	More information about the visual plug-in of translate5 (paid plug-in, only viewable for users with appropriate rights) The contained files are used to generate a layout for visual translation or visual review (with "What you see is what you get"). Possible contents of the directory visual are: ▪ A file "reviewHtml.txt", that contains one or multiple URLs (one URL by line and nothing else). The order of the URLs should be the same as the order of the corresponding files in workfiles. ▪ OR: A number of PDF files OR HTML files (mixing PDF with HTML is currently not supported - please contact translate5s developers, if you need this) ▪ The PDFs / HTMLs contain the same text as the text that is contained in the files in the workfiles directory. ▪ If the workfiles directory contains bilingual files, the PDFs / HTMLs can either be in source or target language ▪ if the PDFs / HTMLs are in the target language, the import option "Connect layout to" should be set to "target" ▪ The order of the PDF / HTML files on the file system should be the same as the order of the contents in the workfiles directory. Yet it is not necessary that the PDFs / HTMLs are the same number of files or have the same file names. ▪ The naming of the files in the visual folder does not matter at all. Only their order should reflect the order of the contents in the workfiles folder. ▪ multiple PDFs are merged to a single PDF and result in a single review file containing several pages/chapters whilst multiple HTML files or downloaded URLs will be accessible through a pager in the frontend ▪ OR: A number of XML/XSL combinations. These combinations will be processed to HTML files which then will be used as the visual source as described above. This incorporates the standard, text-based segmentation and thus these XML/XSLT files are not aligned to the bilingual files in the "workfiles" folder like when adding XML/XSL via the "alignLayout" folder (see below). The stylesheet(s) have to be in the /visual folder as well and must be referenced with a XML preprocessing instruction e.g. <?xml-stylesheet type="text/xsl" href="stylesheet.xsl"?>	visual	no
Directory	visual /image	More information about the visual plug-in of translate5 (paid plug-in, only viewable for users with appropriate rights) Subdirectory in /visual. The contained files/images are used to generate the layout for the visual. You can add one or more images here (file-extensions: *.jpg, *.jpeg, *.gif, *.png, *.bmp, *.gif, *.tiff, *.webp) alongside with an optional web-font (file-extension: *.woff) to generate the visual with OCR-analysis of the image. The web-font will be used for all detected text if given. Please note, that a valid Google Cloud Vision API key is needed for this, see here	visual	no
Directory	visual /video	More information about the visual plug-in of translate5 (paid plug-in, only viewable for users with appropriate rights) Subdirectory in /visual. If a Video is supplied here, a visual video can be generated. Only in conjunction with a workfile with timecodes, that connect the segments to frames in the video, which currently must be a .xslx or .srt file. The prequesites are: • Video: Filetype mp4, codec h264, max. size 250MB, max. Resolution FullHD (1920x1080), max. Bitrate 1MB/s • Excel Workfile: Filetype xslx, 3 columns, first column "Timecode from", second column "Timecode to", third column "Segment text"; The first row may be a header row "from", "to", "text" • SRT Workfile: SubRip subtitle file, filetype srt as defined here: https://en.wikipedia.org/wiki/SubRip • Supported Timecode formats; The timecodes in the workfiles must be in one of the following formats: "HH:MM:SS:FF", "HH:MM:SS,mmm" (srt), "HH:MM:SS.mmm" (vtt) or "v" (plain milliseconds)	visual	no
File	*.xsl	More information about the visual plug-in of translate5 (paid plug-in, only viewable for users with appropriate rights) Xml files in the directories "workfiles" and "alignLayout" can contain an xml-type reference to an xsl file, that is able to transform the XML to a layouted HTML version. This reference to xsl can either be an http(s) link or can be a path relative to the xml file pointing to an xsl file within the zip package	visual	no
Directory	alignLayout	More information about the visual plug-in of translate5 (paid plug-in, only viewable for users with appropriate rights) Similar to the concept of a pivot language it is possible to use an aligned XML with a referenced XSL stylesheet (see above) as source for the visual review. This XML will be aligned to a matching bilingual file in the workfiles folder, meaning that it is expected to hold the same segments (what is validated). The filename of the aligned layout must be identical to the bilingual that contains the same contents, except the fileextension. So "xyz.xml" matches "xyz.xml.sdlxliff" and it also matches "xyz.sdlxliff" and "xyz.xml.mxliff" or "xyz.mxliff". The aligned XML must be provided in a folder "alignLayout" and define a stylesheet via processing instruction as described above. It is possible of using multiple aligned XML/XSLT files for multiple bilingual files in "workfiles". The stylesheet has to be in the same folder and must be referenced with a XML preprocessing instruction e.g. <?xml-stylesheet type="text/xsl" href="stylesheet.xsl"?>	visual	no
File	*.tbx	Contains the terminology for the task. The contained terminology is used to mark terms in the task. Must reside on the top level of the zip-file. Only the first tbx-file in the zip file is used. Please see here for information, which terms are used for a task.		no
File	QM_Subsegment_Issues.xml	Contains the MQM issue types, that should be used for the task. If present, overrides the default MQM issue types for this task. An example file you find here.		no
File	task-template.xml	Contains a task template. Can be used to configure translate5 in an task-specific way.		no
File	DEU-ENG.transitConfig	Only for import packages, that contain STAR Transit NXT files. The first three letters of the file name must contain the transit-specific source-language file extension (in the example "DEU" for German). The last three letters of the file name (before the file extension ".transitConfig") must contain the transit-specific target-language file extension (in the example "ENG" for English). The file itself can be empty. It is a flag file.		no
File	pixel-mapping.xlsx	This optional file can be provided to define the pixel-length for a pixel-based length check in translate5. It defines the pixel-width for each character of a font for a certain font-size.		no

Usage hints

- Folders and files mentioned in the above package structure MUST reside on the top level of the zip file (NOT inside some sub-folder).
- SDLXLIFF, openTM2 XLIFF and CSV-files can be mixed inside the workfiles folders an any needed way. STAR Transit NXT files have to be in a separate import package.
- Please use only ASCII characters in file or directory names and no special characters or German "Umlaute".
  The Zip format does not provide any information in which encoding your operating system has saved a file or directory name. Due to this lack of the ZIP standard, it might cause errors inside translate5, if you do use file or directory names with NON-ASCII characters. Only exception to this rule: Your file system does use UFT-8 encoding (like Linux and Mac do). Addition: In September 2018 a test with Windows 10 showed, that special characters seem to be encoded also as utf-8 under windows with zip. Independent if native windows zip, winrar or 7zip was used. So maybe this issue is solved meanwhile through an always advancing Windows system 🙂