

Technical Support Manual for Translation Editor

Import and Export

Interaction of TE Import and Encoding Conversion

On the Writing System Properties dialog, when you select an encoding converter for a particular writing system, that encoding converter will be used to convert data for importing into Translation Editor if and only if:

- You choose Paratext 5 or Other Standard Format as the source of data for import. (If you are importing directly from Paratext 6, then Paratext takes responsibility for encoding conversion based on the P6 project setup).
- Your data files are not already in Unicode.
- The encoding converter you select is capable of converting from Legacy to Unicode (i.e., Conversion Type must be either **Legacy to Unicode** or **Legacy to and from Unicode**).
- The encoding converter is still available on your computer at the time you import the data. (If the encoding converter is not registered in the repository, TE will display a message indicating the problem. If it is registered but the mapping file needed by the conversion engine is not accessible, TE will import the data without alerting you to the problem, resulting in *legal* but incorrect Unicode characters.

Translation Editor Import

Types of Import Sources

In Step 2 on the **Standard Format Import Setup** dialog box, you indicate the source of the data you wish to import. In most cases, the choice is fairly straightforward, but there are a few issues that might require some clarification. Use this guide to help you select the correct option:

Paratext 6 vs. Paratext 5

Choose **Paratext 6** if the data you wish to import has been edited using Paratext 6 on the computer where you are importing. If the computer does not have Paratext 6 installed or if the project files are from a different computer and have not been copied to the correct folders for Paratext 6 to be able to find them, choose **Paratext 5** instead.

The **Paratext 6** option currently only allows for a single back translation and a single notes (i.e., annotations) project. If you have multiple back translations or multiple projects containing notes (e.g., one with consultant notes and another with translator notes), you should select the **Paratext 5** option.

Note that if you are importing directly from Paratext 6, then Paratext takes responsibility for conversion from legacy encodings based on the P6 project setup. If you are importing from Paratext 5, then TE is responsible for conversion from legacy encodings to Unicode. Therefore, if you have set up your FieldWorks writing system differently from the encoding conversion settings in Paratext 6 (which is probably not usually a good idea),

the choice you make between Paratext 6 and Paratext 5 will affect how the encoding conversion is done.

Paratext 5 vs. Other Standard Format

Both the **Paratext 5** and **Other Standard Format** options require you to select the individual Scripture data files that make up your project. These two options are nearly identical. The difference is that if you choose **Paratext 5**, TE will interpret all backslashes in the data as beginning a SF marker; whereas, if you choose **Other Standard Format**, only backslashes at the beginning of a line will be regarded as the start of a SF marker. This is because Shoebox and Toolbox allow backslashes in the data because they are sometimes needed in filenames (and in one or two instances this ANSI code point has actually been used for a glyph in a vernacular font that). If you choose **Other Standard Format** and have some inline markers that do begin with a backslash, you will need to add those explicitly in the **Mappings** step (Step 4) of the **Standard Format Import Setup** dialog box.

Import Project vs. Single File

One design decision in TE that might be a source of confusion for some users is that the import can be set up for any or all files in a Standard Format project, regardless of which specific files the user might want to import initially. (With the Paratext 6 option, this confusion is avoided since the setup process involves selecting a Paratext project, not the individual files.) Once the import project has been set up to include all the project files, it is not necessary to add or remove files each time the user wants to import. However, if new SF files are created as part of the translation project, it will be necessary to add those files in Step 2 on the **Standard Format Import Setup** dialog box. Likewise, if project files are moved to a different folder location, the files will have to be removed and re-added from their new location. If a SF project includes “template” files (pre-populated files with \id, \c, and \v markers), those files can be added to the list of files in Step 2 on the **Standard Format Import Setup** dialog box in order to make them available for import in the future when those books have been translated. If template files do not exist, a support person can temporarily create empty files following the expected naming convention, add those files to the import project, and then remove them. In the **Import Standard Format** dialog box, the user chooses what to import based on Scripture references (currently constrained to be whole books), not files. In order to import a particular book, the corresponding file(s)¹ must already be added to the import project in the **Standard Format Import Setup** dialog box. TE does not currently have a simple way to select a single file for import.

Interleaved vs. Separate Back Translations and Annotations

If your translation project includes back translation and/or annotation data, you have probably saved this data in one of two ways: either interleaved with the vernacular Scripture (using distinct SF markers) or in separate files.² It is possible to have a hybrid

¹ Books need not be stored in a single file. They can be separated across multiple files (e.g., one chapter per file). A single file can also contain multiple books.

² TE does not currently support importing from the Paratext TNE program.

situation where, for example, your vernacular data has translator notes interleaved and you have separate back translation files with consultant notes interleaved. Whenever multiple types of data are interleaved, the markers must distinguish the type of data as well as the meaning of the data. For example, if your back translation is interleaved with your vernacular Scripture, you obviously cannot use the USFM markers for both. If you prefix the markers used for vernacular fields with \bt, then TE will automatically interpret those fields as containing the corresponding back translations of the vernacular fields. For example, if \p maps to the Paragraph style, then \btp will be the back translation for the corresponding paragraph. With the exception of \v, \vt, \ft and other markers that are mapped to Default Paragraph Characters, it is not necessary to use the \bt prefix for character fields, since these will automatically be included as part of the paragraph in which they occur, whether vernacular or back translation.

Import Mapping Setup

Translation Editor uses styles (similar to MS Word) instead of Standard Format markers to identify different elements of Scripture data. TE has default import mappings for the SF markers in UFSM 2.0 and for a few other common markers. Most users will probably find that the defaults mappings work quite well. If you are setting up mappings for a project (Step 3 on the **Standard Format Import Setup** dialog box), here are a few things you should know:

- TE stores two distinct sets of import mappings:
 - Scripture mappings are used for vernacular and back translation data. If your vernacular Scripture and back translation are not interleaved (i.e., come from different files) you cannot use the same marker to mean one thing in the vernacular but a different thing in the back translation. You can have markers that are used in only one or the other. The set of Scripture mappings will also include any markers for interleaved annotations.
 - Annotation mappings are only used if you are importing non-interleaved annotations. This separate set of mappings allows for the possibility of using the same marker in an annotation file to mean something different from what it means in Scripture.
- **Default Paragraph Characters** is a special character style. The formatting for text marked with this style will be the default formatting for the containing paragraph. During import, the **Default Paragraph Characters** style will be applied to the following types of text:
 - Contents of SF fields explicitly mapped to this style (e.g., \vt).³
 - Any text following a verse number in a \v field.
 - Any text in a field that is mapped to a paragraph style.
 - For inline markers, any data following the end marker (i.e., not otherwise marked with another marker).
- Figures: See Help topic **Invalid figure parameters** for a description of the required format for a marker that is mapped to the **Figure** property.

³ If **Footnotes** is selected in the **Text is part of** box, the data in the SF field will be included in the body of the current footnote. If footnotes are not marked with inline markers, to end a footnote, map the marker to the **Default Paragraph Characters** style and select **Scripture** in the **Text is part of** box.

- Writing systems: Scripture data is usually imported using the default vernacular writing system specified for the FieldWorks project. Back translations and annotations usually use the default analysis writing system. If the Standard Format data contains fields in a specific writing system other than the default for the context, you can override the default writing systems. If you don't need to override the defaults, it is best to leave them unchanged. If there are separate project files for back translations or annotations in each distinct writing system, you can select the appropriate writing system on the **Back Translation** or **Annotations** tab in Step 2 on the **Standard Format Import Setup** dialog box. If a single file contains multiple writing systems, distinct SF markers should be used for data in each writing system. In the Modify Mapping dialog, select the appropriate writing system for any marker for which the default writing system is incorrect.
- Unmapped markers
 - Omissions you should add to TeStyles.xml (these accidental omissions will be corrected in the next version of TE):
 - \ft Default Footnote Characters
 - \ie Excluded
 - \xt Default Footnote Characters
 - Any markers that are not part of the USFM 2.0 standard⁴ or one of the widely used standard sets of mappings that TE knows about need to be mapped manually. If you encounter unmapped markers that are part of a branch standard or in widespread use, you can consider adding them to TeStyles.xml. You might also send an e-mail to TEAnalyst@sil.org to ask that these markers be considered for inclusion in the standard set in a future version of TE. During import, if TE encounters any markers in the Scripture data that are not mapped (whether automatically or manually), it will create a new paragraph style whose name is the SF marker. This prevents unintentional data loss and permits the user to clean up the data in TE after importing.
- Inline markers: If you are importing from Paratext (version 5 or 6), TE will detect inline markers (using the form `\abc...\abc*`) automatically. You cannot use any other format for inline markers for Paratext data (**Add** is unavailable in Step 4 on the **Standard Format Import Setup** dialog box). If your import source is **Other Standard Format**, then you must manually define any inline markers used in your data. Inline markers can begin with a backslash (\), but they are not required to. See Help topic **Inline markers** for complete rules for defining inline markers.
- Unsupported markers (not an exhaustive list)
 - Advanced footnote markup: \fl, \fm, \fp
 - Advanced introductory markers: \imt4, \io, \io4, \iq3, \is2
 - Chapter Label: \cl maps automatically, but is currently not stored in TE
 - Deprecated/Obsolete markers: \phi, \ps, \psi
 - Deuterocanonical material: \dc, \fdc, \xdc

⁴ Official documentation of the USFM standard is available on-line at <http://confluence.ubscap.org/display/USFM/Home>

- Front/back matter: \conc, \cov, \fe, \glo, \idx, \intro, \k1, \k2, \maps, \ndx, \p1, \p2, \pref, \pub, \toc, \wg, \wh, \wr
- Hard formatting: \bd, \bdi, \em, \it
- Publishing & meta-data markers: \cp, \h1, \h2, \h3, \pb, \pubinfo, \restore, \spine, \toc, \toc1, \toc2, \toc3, \vp
- Study Bible notes: \ef, \efm
- Tables: \tr1, \tr2 (*Note:* \tr maps automatically to Table Row style, but TE does not yet support table formatting)
- Miscellaneous: \addpn, \cd, \li4, \ms2, \ms3, \mt4, \mte1, \mte2, \nb, \pi3, \pro, \q4, \qac, \s3, \s4, \sls, \sr

Tips and Tricks for Importing Footnotes

TE attempts to interpret several different formats of footnotes (in addition to the two approaches described in the USFM documentation). If the default import mappings don't work for a particular format of footnotes, you can either convert the data to conform to the USFM 2.0 standard or try to change the import mappings to work with the data. If you need to change import mappings for footnote-related Standard Format markers, observe these guidelines:

- When modifying the mapping for SF markers that indicate the beginning of a footnote or part of the footnote body, select **Footnotes** in the **Text is part of** area on the **Modify Mapping** dialog box.
- When modifying the mapping for inline markers that can be used in either Scripture or the footnote body (e.g., the Emphasis style), select **Scripture** in the **Text is part of** area on the **Modify Mapping** dialog box.
- Some markup formats don't include a separate marker that indicates the start of a footnote. A single marker indicates the element of the footnote body and at the same time implies that a footnote is starting. If this marker designates something other than plain text in the footnote body, you should map it to the appropriate specific style (e.g., **Note Target Reference**). TE will automatically create a new footnote and assign the paragraph style **Note General Paragraph** to the footnote body.⁵
- If the imported data includes the footnote caller in a separate Standard Format field, exclude this field from import and use the **Footnotes** tab of the **Scripture Properties** dialog box to specify which type of callers to use.
- If the imported data includes the footnote caller at the start of the field which begins the footnote (e.g., at the beginning of the \f field, as per the USFM standard), map that field normally without attempting to account for the caller. TE will attempt to detect the caller and discard it. The current version of TE does not detect and discard literal callers consisting of an alphabetic character, so you would need to clean that up manually.
- If you have never inserted a footnote in TE nor changed the footnote properties from the default settings, then the first time you import data containing footnotes or cross references, TE will analyze the data and attempt to set those properties

⁵ This markup format should not be used if it is necessary to distinguish between general and cross-reference footnotes.

appropriately. If TE misinterprets the data, you can easily change these settings after importing.

Finally, it should be noted that TE does not currently support any of the following:

- Markup where the footnote caller is specified using an inline marker but the footnote body follows later in fields that come after the end of the paragraph or verse.
- Separate import of cross-reference footnotes. Some branches, such as PNG use a standard set of cross-references that can be included during the publishing process. Currently, these would need to be added manually in TE.
- Footnotes in annotations.

Import of Key Term Renderings

Currently TE cannot import data files containing key terms and their vernacular equivalents. This is planned for some future version (sample data from the field is welcome).

Translation Editor Export

When you export data from TE, the current book filter is always used to determine what will be exported. Following are brief descriptions and notes concerning the various export formats supported by TE.

Document (RTF)

Use this format to export your vernacular Scripture data into Rich Text Format. This is most suitable for getting your data directly into Microsoft Word or another word processing program that can interpret RTF. This can enable you to format your Scripture for printing in ways that TE doesn't currently support (such as multiple-column or booklet printing). This format does not include annotations or back translations and should not be used for long-term archival.

Standard Format (USFM)

Use this format to export your data in a format suitable for automated Scripture checking, sharing via e-mail, data processing, or editing on computers that do not have direct access to the FieldWorks database. The TE team is aware of a couple minor issues where our exported markup might not conform to the best practices for USFM data. We are in close communication with IPUB to identify and fix any problems, but we welcome input from the field as well.

USFM-Paratext

When you export to this format, you can export either your vernacular Scripture or a back translation. If you want both (or if you have back translations in more than one language), you will need to export them separately because Paratext only supports one language per project.

TE automatically generates a Paratext stylesheet (.STY) when you export your data. This stylesheet includes all the information from the standard USFM stylesheet, but the formatting details will reflect the information in the TE stylesheet. Also any markers that

are included in the export from TE that are not part of the USFM standard will also be included in this generated stylesheet. For the purpose of Scripture checking or final publishing, you should usually use the standard usfm.sty rather than the generated stylesheet, but if your intention is to see your data in Paratext formatted as much as possible the way it is in TE, use the generated one. TE attempts to create the generated stylesheet in your My Paratext Projects folder, naming it the same as your FieldWorks project (but with an extension of STY.). If the folder is not accessible, TE will display an error message and the stylesheet will not be created, but the data will still be exported. We expect that in a future version of TE some of the details will have been worked out with IPUB to provide a clearer approach to checking and publishing, given the mismatches between the TE stylesheet and the USFM standard.

To open exported data files in Paratext for checking or publishing, do the following:

1. Create a project in Paratext (use the latest usfm.sty).
2. Choose the language and font that matches the Project Properties in TE (note that if you make the abbreviation the same in both tools it works better).
3. If you have pictures, see the section titled Publishing Pictures.
4. Import the exported data into Paratext (unnecessary if you just export it to the correct location in My Paratext Projects).

USFM-Toolbox

The Toolbox export does not attempt to create any of the Toolbox metadata files (.LNG, .TYP, .PRJ). It also does not write the Toolbox-specific markers at the beginning of the file. When you open the exported files in Toolbox, all that will be taken care of.

The Toolbox USFM export differs from the Paratext export in a few ways:

- It includes \rcrd as the record marker.
- It includes \vref as a marker for each verse so that Toolbox can synchronized scrolling and referenced-based “jumping.”
- It includes \vt fields with the verse text (distinct from the \v field that contains the verse numbers).
- It uses a vertical bar character (|) instead of a backslash (\) to designate the start of an inline marker.⁶
- It includes the option of exporting back translations interleaved with your Scripture data.

Publishing Pictures

Since releasing TE version 1.3, we became aware that exporting pictures with a full path poses a problem for Paratext Scripture checks and Publishing Assistant. This makes it difficult to publish TE data using InDesign. The current work-around is to search for each occurrence of \fig in the exported data in order to remove the path specifications and add the end marker \fig* manually, and then copy the picture files from Program Files\FieldWorks\Pictures to the **Figures** folder in your Paratext project folder. Before long, someone might create a CC table to make this easier, but we hope to analyze and fix this problem in a stable release scheduled for December 2006.

⁶ Some research might be needed to determine if this was the right approach.

There may also be issues with the exported format of pictures in Toolbox data. As far as we know, Toolbox doesn't do anything directly with pictures, but we aren't sure what it does, for example, when it exports to RTF. In addition, there might be external processes that expect a certain format for figure fields coming from Toolbox. More research (and field input) is needed here.

Data (XML)

We discourage you from using the current XML export because it is incomplete and we plan to change it to conform to the OSIS best practices standard.

Roundtrip Import and Export Issues

There are several issues which pose problems for safe (lossless) roundtrip importing and exporting of data between TE and Standard Format.

Issues Likely to be Resolved in Future Versions of TE

We cannot commit to an exact timetable, but we will try to deal with these issues based on field feedback. These are approximately in decreasing order of priority based on our current understanding of the need:

- The current version of TE requires a full for image files and also omits the end marker `\fig*`, both of which are failures to conform to the USFM 2.0 specification. Manual intervention is required to add or remove the path in order to go back and forth.
- The current version of TE can't export 1 chapter per file, so if your imported data was stored that way, the exported data will be different.
- TE does not attempt to detect re-imported annotations. If you re-import data containing translator or consultant notes that are already in your TE project, TE will create duplicates. In addition, TE has a much richer model for storing annotations than USFM, so exported annotations will lose information.
- If you export to Toolbox format, TE will use a vertical bar character (|) to designate the start of an inline marker. If these markers were not part of your original markup, in order to re-import this data correctly into TE, you will need to add import mappings for these new markers. TE does not automatically detect inline markers in non-Paratext data. We need to determine whether this is the best way to mark up data for Toolbox and, if so, implement a strategy that allows you to easily re-import this data.
- There are 12 built-in styles that you can apply in TE that do not have corresponding USFM markers: Abbreviation, Alluded Text, Attribution, Gloss, Label, Mentioned, So Called, Speech Line1, Speech Line2, Variant, Variant Paragraph, Variant Section Head. In addition, users can add other user-defined styles which will not have USFM equivalents (although this is not encouraged). If any of these styles are used, the exported files will include markers (corresponding to the TE style name) that are not part of USFM. The exported data can be re-imported into TE without loss, but programs expecting well-formed USFM data might not work properly.

- TE can import annotations from a separate Paratext project, but if you export annotations from TE to Paratext, they will be interleaved.

Issues not Likely to be Resolved in Future Versions of TE

Some round-trip issues will probably not be resolved for reason or another. Some may be by design, in order to promote our corporate directions. Others may be inherent to the distinction between TE and Standard Format.

- Mappings between TE Styles and SF markers which are not 1-1. For example:

TE Style(s)	USFM marker(s)
Stanza Break	\b ⁷ , \ib
Emphasis, Foreign, Supplied, etc.	\bd, \bdi, \em, \it ⁸
Alternate Reading	\fq, \fqa, \xq
Intro Section Head	\imte, \iot, \is, \isl
Intro Citation Line1	\iq, \iq1, \ipq
Label, List Item1	\li, \li, \ph, \ph1
Title Main	\mt, \mt1, \imt, \imt1
Section Head	\s, \s1, \mte

- Some previous versions of TE erroneously exported footnotes with an \ft* marker (which is not part of the USFM specification). If you wish to re-import data exported from a version of TE that had this problem, the \ft* marker should be manually removed first. If not, TE will insert a paragraph break after the footnote marker and apply a new style called \ft*.
- TE supports the Unicode hard line break character. Since this character is not supported by Paratext or Toolbox, it is stripped out during export.
- Title Secondary and Title Tertiary are character styles in TE but paragraph styles in USFM 2.0. TE can use hard line breaks to format secondary or tertiary titles such that they appear as separate paragraphs when this is desirable. Since these styles are they are always exported as paragraph markers, if they have been applied in TE to text that is on the same line as the main title, the exported data will not reflect this. Therefore, when they are re-imported, they will they will always be formatted using hard line breaks.
- TE only exports Unicode data, so if you begin with data in a legacy encoding, manual intervention will be required after export to get the data back to its original state.
- TE exports data that conforms to the USFM 2.0 standard. If your incoming data does not conform to the USFM 2.0 specification, the markup of the exported data could be significantly different from the original. A future version of TE might make it possible to control export mappings if there proves to be a need for it.

⁷ The marker \b has sometimes been used in SF data to produce extra space between paragraphs where a Stanza Break is not appropriate.

⁸ The markers \bd, \bdi, and \it have sometimes been used in SF data to achieve hard formatting corresponding to multiple functions.

- The USFM standard provides for two different (but equivalent) ways of formatting footnotes, nested and non-nested. TE exports using the non-nested format, so if the incoming data used the nested format, they will be different in that way.