

Technical Support Manual for Customizing External Control Files for Translation Editor

Customizing the TE Stylesheet

TE uses a control file to generate built-in styles and control import and export mappings. This control file is Program Files\FieldWorks\Translation Editor\TeStyles.xml. The corresponding DTD is Program Files\FieldWorks\styles.dtd.

Understanding the Different Sections of Stylesheet

TeStyles.xml has three major sections:

Styles

The first section contains the definitions for the built-in styles used in TE. For each style, there is information that tells TE where and how the style can be used; an export mapping, if appropriate; a short description to tell the user how to use the style; and specification of the presentation characteristics for the style (e.g., font size, indentation, etc.). This list was produced in consultation with IPUB and has been approved by them for use in Scripture publications.

Replacements

The second section of TeStyles.xml lists replacements that TE will use to rename or merge styles when upgrading. If a built-in style is being renamed or deprecated, an entry for it will be in this list. Once added to this list, an entry will not be removed. This makes it possible to upgrade from any previous version of the stylesheet without custom logic that knows about the specific version (which is important since it might be upgrading from a field-customized version). You will not usually customize the <replacements> section of TeStyles.xml, except to replace or rename custom styles that you added previously.

Import Mappings

The final section of TeStyles.xml is a list of import mappings from Standard Format to TE styles. In the future when TE supports multiple import settings, it will be possible to add additional lists of import mappings, but for now there should only be one list, called **TE Default**. If you are customizing the import mappings, take care not to delete or rename this list.

When and Why to Customize

Occasionally, you might need to customize TeStyles.xml for one of the following reasons:

- If a branch has a need for "standard" styles which are not built in. (If you add styles, make sure you understand the implications for later publishing.)
- If a branch is following a standard other than USFM 2.0 for Scripture markup, you can customize the import mappings so that each project will not need to modify the import mappings.

- If a project is using non-standard styles, they can be added using the **Style** dialog box. However, there is currently no place in the TE user interface to set export mappings, so it might be necessary to add the styles to TeStyles.xml to specify the correct SF markers to use for export.
- If branch or local publishing guidelines call for a different look (e.g., section heads should use a sans-serif font), the presentation characteristics of the built-in styles can be modified to match those guidelines.
- Localization of usage info (The localization can be done now though it can not yet be made visible in the UI in TE)

Note that there is no need to delete styles to limit the length of the list of styles on the toolbar. To limit the list to a smaller subset, in the **Options** dialog box, select the Styles tab, and then choose which levels of styles to display.

Setting the Stylesheet Version

Whenever customizing TeStyles.xml to add, delete or change styles, you must change the markup version. Each time TE opens a database, it checks to see whether the stylesheet version stored in the database matches the current markup version in TeStyles.xml. If it does not, TE reads the styles from TeStyles.xml and applies any changes to the stylesheet in the database. The markup version is a globally unique identifier (GUID), rather than a sequential number. This makes it possible for TE to distinguish between any two versions without the need for someone to handle global coordination of version numbering (i.e., a customized version created in PNG will never have the same version as a new version created by the TE team. To generate a new GUID, use the program guidgen.exe or visit www.guidgen.com. Note that you do not need to change the version number if you are merely adding or changing default import or export mappings.

Reserved and Internal Styles

There are several styles that TE requires to function properly. Some of these are internal styles that the user can not apply directly. Others are default styles used for newly inserted paragraphs in a particular context. If you accidentally delete one of these styles from TeStyles.xml or change its behavioral attributes (e.g., context, structure, type, etc.) in a way that is inconsistent with TE's expectations, TE will override your edits for those styles.

Understanding the Style Hierarchy

The TE Stylesheet uses a hierarchy to allow characteristics to be inherited. Before making a change to the appearance of a built-in style, make sure you understand its place in the hierarchy to be sure you are applying the changes at the correct level. For example, if you want book titles to use a sans-serif font, you would set the font type to sans for the **Title Main** style. But if you want all headings to use a sans-serif font, you would set the font type to sans for the **Section Head** style, since it is the basis for all the section head paragraph styles and **Title Main**.

Validation

After customizing any XML file, you should validate it against the DTD file before attempting to use it in the program. There are several different validation tools available. If you use Internet Explorer, there is an XML validation plug-in available for free from Microsoft. You can download iexmltls.exe from MS downloads.

Communicating Changes to IPUB/TE Team

If you find that you need to add or make significant modifications to the standard built-in set of styles, it would be wise to communicate with the TE team and IPUB to describe the reason the built-in stylesheet did not meet your needs. This will not only give us feedback to help us improve our software, it will also give us a chance to help you head off potential problems or find other solutions to meet your needs.

Redistribution of Customized Stylesheets

We have discussed the possibility of adding branch-standard customizations to our official distribution. If you have made customizations and you would rather not have to administer the distribution to your branch members, please let us know and we can discuss this further.

Customizing the TE Key Terms List

The list of New Testament key terms is initially populated the first time you open a project in TE. The key term information consists of a hierarchy to provide logical organization of related key terms and the concordance information that indicates where in Scripture each key term occurs. The key terms file is Program Files\FieldWorks\TeKeyTerms.xml. The corresponding DTD is Program Files\FieldWorks\KeyTerms.dtd.

When and Why to Customize

There are several reasons why it might be desirable to customize the built-in list of key terms:

- Compliance with standards already in use in your branch
- Localization
- Additional key terms (e.g., Old Testament, proper names, words that are significant in receptor language/culture, etc.).
- Alternate hierarchy preferences
 - Flat list
 - Further splitting (e.g., older brother vs. younger brother)

Limitations on Customized Key Terms Lists

There are some significant limitations in TE's support for customized key terms lists:

- TE reads an XML file to generate this list, so if you need to customize the list, you must replace the standard file with your customized file before creating your project or before opening it for the first time in TE.

- If you customize your key terms list but wish to merge it later with an updated standard list of key terms distributed with TE, it can be difficult since the hierarchy might not match. Our plan is to use the Romanized transliteration of the original Biblical term as the “key” field for finding matches between two different lists when upgrading. However, since the customization strategy has not been fully implemented, no testing has yet been done.
- There is not yet any way to work with the list of key terms in a GUI. It’s fairly easy to make mistakes using a text or XML editor.

Because of these limitations, we request that you contact the TE team before customizing the list so we can make sure your intended customizations are in keeping with our planned strategy. Also, we would like a final copy of your customized list so we can attempt to ensure that it is covered when our merging strategy is in place.

Customizing the Publications Configuration File

TE has three different built-in print layout views. Each one has its own page settings (paper size, margin widths, etc.). To view or change the page settings for the active print layout view in TE, on the **File** menu, click **Page Setup**. The default page settings are initially populated the first time you open a project in TE. The publications settings file is Program Files\FieldWorks\Translation Editor\TePublications.xml. The corresponding DTD is Program Files\FieldWorks\Translation Editor\TePublications.dtd.

When and Why to Customize

There are several reasons why it might be desirable to customize the built-in publication settings:

- To use the local standard paper size (e.g., A4).
- To make Metric units the default.
- To choose header/footer elements to match branch standards.

Note that you should not bother customizing TePublications.xml merely to change the settings for a single project. Instead, use the **Page Setup** dialog box in TE.

Limitations on Customized Publication Settings

There are some significant limitations in TE’s support for customized publication settings:

- As with the key terms control file, if you need to customize the publication settings, you must replace the standard file with your customized file before creating your project or before opening it for the first time in TE.
- You can change the settings for an existing publication in TePublications.xml, but you should not delete or rename the existing publications or header/footer sets.

Customizing the Scripture Annotation Categories File

When you create annotations (Translator or Consultant notes) in TE, you can assign standard CONNOT categories to them. TE initially populates the hierarchy of Scripture annotation categories the first time you open a project in TE. The annotations categories file is Program Files\FieldWorks\Translation Editor\TeScrNoteCategories.xml. The corresponding DTD is Program Files\FieldWorks\Translation Editor\TeScrNoteCategories.dtd.

When and Why to Customize

The CONNOT categories are a widely used standard which will probably facilitate data sharing, but if your branch wishes to use a different system for categorizing your annotations, you can customize or replace the list.

Use of Topics List Editor

Unlike the key terms list and publication settings, the annotation categories can be edited easily in FieldWorks Topics List Editor for a particular project. Unfortunately, to redistribute this list for use throughout a branch, you will need to export it in Standard Format and come up with some process (manual?) to reformat it as an XML file that conforms to the DTD.