

The Translation-Friendly Template

Minimizing Localization Issues in FrameMaker Templates

Scriptorium Publishing Services, Inc.

P.O. Box 12761, Research Triangle Park, NC 27709

919-481-2701 or sales@scriptorium.com

<http://www.scriptorium.com>



In today's global economy, the translation of technical documents into other languages—called *localization*—has become necessary for many companies. Too often, localization becomes an expensive, unending process. The translation itself is costly, and after translation, companies spend astonishing amounts of time and money preparing translated files for printing.

It's unlikely that your finalized English FrameMaker files will survive the translation process and emerge production-ready in your target languages. Because the translated counterparts of most English words are longer, the text expands, undoing much of the production work you did in the English files—for example, it's unlikely that the careful copyfitting in your original files will be usable in the translated version.

You can't completely eliminate these problems. However, as this document explains, you can design your FrameMaker templates to minimize common localization issues.

Handling pesky page breaks

When cleaning up English files for production, you occasionally have to force page breaks by using paragraph overrides. These custom page breaks rarely achieve the desired effect in translated files because of the expansion of text—instead of occurring at the end of a page, a break might occur halfway down a page. To correct these page break problems, you can import formats to remove the manual page breaks and then insert new page breaks where needed. But there is another option worth evaluating.

In the Paragraph Designer, you can specify that certain heading levels always begin at the top of a page. Suppose that a Heading2 paragraph tag denotes separate sections within a chapter. (Heading1 format is the title of the chapter.) You could specify that the Heading2 format always starts at the top of a page by selecting *Top of Page* from the Start drop-down list on the Paragraph Designer's *Pagination* tab. This solution does not eliminate problems with page breaks completely, but it certainly will minimize them.

From a design perspective, setting up a particular heading level to start at the top of a page is probably not the best choice, and the time saved is small if you are translating to just one language. However, the benefit of making such a compromise becomes

much more apparent when translating to several languages—the savings in copyfitting time are multiplied by the number of target languages.

The invisible table: friend and foe

Using invisible tables is a double-edged sword when it comes to translating FrameMaker files. They can dramatically reduce the amount of work you do alphabetizing glossary entries, but they can also create a lot of unnecessary work when used to create notes and warnings.

Be sure to allow for text expansion when setting up any table. Too much text squeezed into a table cell can cause unwanted wrapping, as shown in the examples in “The unfriendly invisible table” on page 4.

Alphabetizing glossaries the easy way

Invisible tables can be very helpful when dealing with translated glossaries (and other alphabetized lists). Regardless of the target language, a translated glossary won't be in alphabetical order just because the English source was. Generally, the glossary must be realphabetized manually.

However, you can avoid this chore by creating a one-column invisible table that is the width of the text frame and placing each glossary entry (term *and* definition together, that is) within a cell of that table. Then use FrameMaker's new table sorting feature¹ to alphabetize the translated entries. The laborious process of cutting and pasting entries is replaced with a few clicks of the mouse: select *Table > Sort*, then specify you want to sort the table alphabetically. The table is alphabetized according to the first letters of each entry's term.

You can achieve the same effect by converting your glossary to a table, sorting, and then converting back to text. You could also use a two-column table—one column for the term and the other for the definition—and sort based on the first column. But putting the term in the first cell can cause table sizing problems, as described in the next section.

1. Table sorting was added in FrameMaker 5.5.

The unfriendly invisible table

Some companies use two-column tables without borders to create notes, warnings, and similar tags, as shown in the following figure:

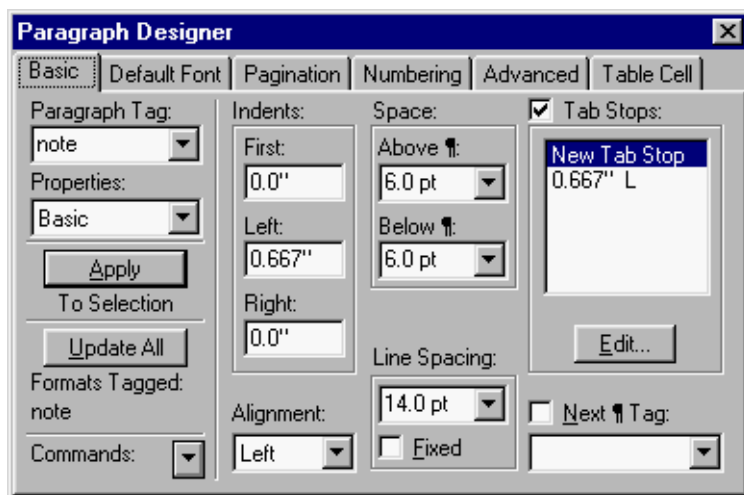
Note: §	This is the body of a note. The use of two columns prevents the text of the note from wrapping underneath the "Note:" tag §
----------------	---

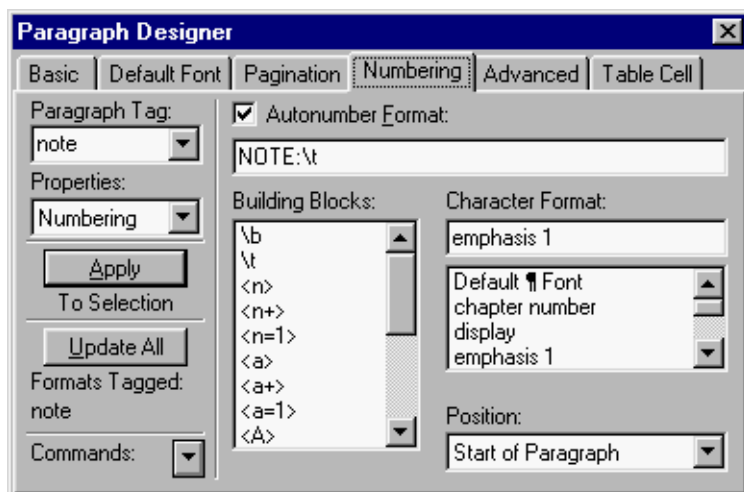
After translation, this note doesn't look very good:

Remarque: §	When the first column of this note table is translated into French, the "Note" tag no longer fits in the table cell. The width of that cell must then be adjusted for each table §
--------------------	--

Because the word "Remarque" is much longer than "Note," the width of every table cell containing "Remarque" must be adjusted, and there is no global way to do this in FrameMaker.

A better solution is to develop a paragraph format that uses auto-numbering to create the note format. To prevent text from wrapping underneath the "Note:" tag, set up tabs in the Basic settings of the Paragraph Designer. The following figures show the basic and numbering settings for this format.





The word “Note” is translated in the format's autonumbering properties, and then you adjust the indents on the *Basic* tab to accommodate the expanded translated word. Selecting *Update All* applies these changes across one file, and the paragraph formats of that file can be imported into all other files to make the change throughout the book. Importing formats in this manner is much faster than adjusting table widths, one at a time, across several files.

The case of the expanding callouts

When creating standards and formats for callouts and other text in figures, it is crucial to keep the expansion of translated text in mind. If callouts barely fit in your English graphics, they certainly won't fit if they expand during translation. Therefore, it's essential that callouts not be crammed into tight spaces.

Another big consideration when handling callouts is how they are created. Some companies choose to insert callouts while creating figures in a graphics program—the callouts are part of the graphic file imported into FrameMaker. However, this means that translators cannot translate the callouts by just working in the FrameMaker source (or MIF files created from the source). The figure must be opened in the appropriate graphics program and the callouts translated, therefore adding another layer of work to the localization process.

Assuming that the only items in a figure that require translation are callouts (that is, the graphic itself does not require localization), importing the graphic into the FrameMaker files and then inserting callouts is a better solution—writers or illustrators add

the callouts within the anchored frames holding the figures. Translators see the callouts and can make the necessary changes in the source. The use of a graphics program is unnecessary.

Regardless of the method you use to insert callouts, consider the size of the type used to create callout formats, and be sure writers and illustrators are aware that translated callouts need more space than their English counterparts. You can document the specifics of callout formats, placement, and so on, within your template. In fact, it's generally good practice to use a template (FrameMaker or otherwise) to explain proper usage of all formats.

Clean source = less work on translated files

Although it's not necessarily a template issue, taking a second look at source files is a good idea. Catching a missing index range marker and fixing it once is much better than correcting the same error in multiple translated versions, particularly if you must pay translators more money to help you fix such mistakes. Taking some extra time to proof source before shipping to translators is a wise investment.

By double-checking your files for errors and developing a template that can better handle the expansion of translated text, you will reduce the localization burden. Even if you don't think your company plans to translate documents, the template concepts described here reflect optimal usage of FrameMaker's features, so it can't hurt to apply them. And if one day you discover that localization is in your very immediate future, you've already taken steps to make that process run more smoothly.

Contacting us

If you have any questions about Scriptorium Publishing Services, Inc., contact:

Scriptorium Publishing Services, Inc.
P.O. Box 12761
Research Triangle Park, NC 27709
919-481-2701
sales@scriptorium.com
<http://www.scriptorium.com>