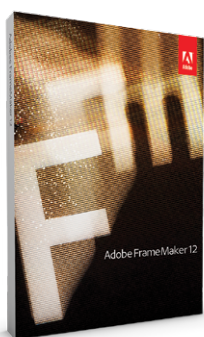


Adobe FrameMaker and XML: Reducing Translation and Publishing Costs



Test drive Adobe FrameMaker 12

Try the full functionality of FrameMaker 12 (as a part of Adobe Technical Communication Suite 5 software) in minutes and without downloading the software. Tutorials are also included. Test-drive it now at www.runware.com/clients/adobe/techsuite.

Executive summary

It has long been known that converting documents and content to structure, especially XML and DITA, can reduce translation costs due to content reuse and other factors. In this paper, Adobe product evangelist, Maxwell Hoffmann, shares his personal experiences from several years of multilingual production with FrameMaker and other products. He outlines major areas where FrameMaker can reduce localization project costs.

Cost savings occur in two general areas:

1. Reduced effort during content creation on the **client's** part
2. Significantly reduced billable time from the **Language Service Provider (LSP)** for:
 - a. Post-linguistic format correction
 - b. Linguistic review cycle
 - c. Swifter edits during Linguistic Quality Assurance (LQA)

While this paper will review the general benefits of XML and DITA for content that will translate more effectively, the author will also touch on new features in FrameMaker 12 that further reduce labor and effort during authoring and post-linguistic content correction.

General benefits of XML for cost-effective localization and translation

For decades, many enterprises have used "flat" document structure in products like Microsoft Word. When content is converted to XML structure, it is possible to assemble "documents" from sub-document structure, like topics and concepts. DITA (Darwin Information Type Architecture) is a general standard that is particularly good at this process.

The following are widely recognized benefits of XML for content that is translated and localized:

- **Separation of content from formatting**—styles, fonts, etc., either come from external style sheets or a component of the publishing solution separate from the document content. Formatting is automatically invoked when elements (e.g. a "list") are selected.
- **Context-sensitive formatting**—promoting or demoting lists and other elements to higher or lower levels can automatically change formatting (e.g. a main list with bullets will change to dashed prefixes if the list becomes a sublist).
- **Consistent document structure**—the number of editing and formatting decisions made by LSPs are dramatically reduced when XML is employed to enforce structure and limit content choices.
- **Content reuse**—XML or DITA projects may be assembled via "maps" with links to a content repository. Basically, the author "writes once" and "uses many times." If one instance of a Caution or Warning is linked 200 times in a project, it only needs to be edited or translated once, and all instances are instantly corrected.

- **Potential reduced learning curve**—staff who do basic authoring or editing do not need to master every nuance of DITA, XML, attributes, etc. Because XML and DITA both reuse the same element (e.g. [title].) in context, there are fewer elements to remember than paragraphs associated with traditional desktop publishing.
 - From personal observations, I have seen several localization clients convert from Word or regular FrameMaker to structured FrameMaker/XML. In many cases, the client was using over 120 paragraph format tags before conversion and about 30 common element tags after migration to XML.
- **Less painful migration path for legacy content**—once documentation assets are converted to XML or DITA, the structure is persistent and less subject to change. You do not have to worry about upgrading a Word 95 document to Word 2013 with some potential loss of formatting.
- **Intelligent metadata**—XML allows custom value for “attributes” associated with elements. For example, a character level element like [code] could contain a language attribute of “EN” for English to indicate that this string of text should not be translated.
- **Structure can be manipulated and re-expressed more easily**—XSLT and scripting make it relatively easy to re-sort or reorder certain types of data. It is not uncommon to require a change to currency or date formats after translation. This type of conversion does not ordinarily happen automatically as part of the translation process.
- **Eliminate the need to “master” a customer’s style guide**—with traditional publishing, LSP publishing staff had to master or memorize the arcane rules and progression of formatting established by each customer. Since XML solutions automatically insert formatting with specific elements, LSP staff have less to learn or remember when moving from one customer project to another.
- **Quicker ramp-up time for contractors and temporary staff**—during a major product release, many customers have additional staff work on documentation who may not have seen it for 9 to 18 months. When documentation assets are in XML, there is less to learn, and it is literally harder to make a mistake.
- **Unwanted format overrides are eliminated**—some authors or publishers who are used to Microsoft Word or older versions of FrameMaker are used to modifying line spacing even point sizes to “make text fit” on a page. This can wreak havoc with translated content. XML and DITA will only permit predefined structural elements to change formatting, e.g. [b] for bold and [i] for italic. Authors may no longer randomly insert font overrides.

How FrameMaker makes XML easier to use

FrameMaker is generally acknowledged as the most widely used tool of choice by teams who create complex, high-volume documentation or tech comm content. Regular or unstructured FrameMaker files can convert into XML within FrameMaker with relative ease. Simple conversion tables and other utilities provided out-of-the-box with FrameMaker reduce consulting time and costs. Our AdobeTCS division has a series of recorded webinars which document every step required in this process.

Simplified authoring model—FrameMaker is one of the few XML authoring solutions that enable the author to view actual page breaks (if print is the final output) during authoring and editing. Since many authors prefer a “content” view that doesn’t display headers and footers, FrameMaker provides an authoring view that resembles more common tools.

Elements may be displayed with temporary tags to make insertion points more evident. A structure view window displays whether the document structure is valid or not.

FrameMaker 12 has introduced a quick element toolbar that enables authors new to XML to insert common elements like tables, lists, etc., without locating a tag name from a catalog. With 20 minutes of instruction, it would be possible to prepare an author to create volumes of certain repetitious content structures.

A personal case study

In one of my last jobs while in the translation industry, I was involved in a project that really dramatized the cost savings that FrameMaker can occur. A state government agency had decided to migrate from Word to structured FrameMaker to create consistent training materials that would enable widespread content reuse.

A consultant cleverly constructed a “customized” DITA application that replaced common element names like [title] [list] [item] with self-document “purpose” names like [course_objectives], [materials_list], [material] and [summary]. I was contracted to provide the training on both structured FrameMaker and DITA over a five day period. The class composed highly intelligent people who had only used Word and had literally never seen FrameMaker.

I was somewhat worried at the lack of questions by Thursday. Was everyone in the class “getting it?” My fears were unnecessary. The consultant who has constructed the structured application had made the available elements so logical and obvious that the entire process seemed familiar to the course developers.

As a precaution, my translation agency created a purchase order for 40 hours of post-training “help desk” support, to be provided personally by me. Over the next nine months, this customer only used 12 hours of Help Desk support. They were able to resolve all other issues on their own.

What is so amazing about this case study is that nearly everyone involved on the client side was new to FrameMaker. Perhaps their openness and lack of preconceived notions made the process go more smoothly.

By the way, this client documented that they eliminated about 75% of cost overruns on projects by (a) enforcing consistent structure and (b) eliminating the possibility of introducing format overrides.

Relevant time-saving features in FrameMaker 12

FrameMaker has recently introduced a number of improvements that are particularly beneficial to anyone involved in content editing and formatting after translation. The benefits apply to customers and LSPs alike. These benefits are not listed in order of importance, and it should be noted that FrameMaker has many additional benefits that are not relevant to the topic at hand.

Multiviews: Native XML Code, WYSIWYG and Author View

Easily work with XML content in Native XML Code View, using customized toolbars, inline attribute editor, error tracking, search and more. Work with styles, themes, and formats in the WYSIWYG View and see content as your users will. Use Author View to focus on writing without bothering about formatting.

WYSIWYG view with optional tags	XML Code View of same content
	<pre data-bbox="1015 1459 1510 1669"><topic id = "id141CL900WZL"><title>Getting Started with F <shortdesc>A simple introduction to working with XML</sho <prolog><author>Maxwell Hoffmann</author></prolog> <body><p>Simply type over the gray bands of text in the e document</p> <p>first item</p> <p>second item</p> <p>third item</p> <p>fourth and final item</p></body> <related-links></related-links></topic></pre>

Customizable workspaces

FrameMaker has specific workspaces for various tasks, like working with unstructured content, or with XML. Users can easily customize their workspaces, and create new ones, that have just the commands needed for the task at hand.

Quick Element toolbar

Simply click on icons in the native Quick Element toolbar to insert or wrap content into elements without requiring knowledge of tag names. The toolbar guides you to select valid elements in the XML structure. Build your own toolbar for custom Document Type Definition (DTDs) by mapping different elements to icons in the toolbar.



Expandable DITA map topics for quick viewing of content

Expand DITA maps by resolving each topicref to easily preview the topic content. Open topics individually or all together at one go. Preview nested topics without opening each topic individually. Edit a topic by simply clicking on it, and refresh to see the changes.

Single-click session restore for quick start

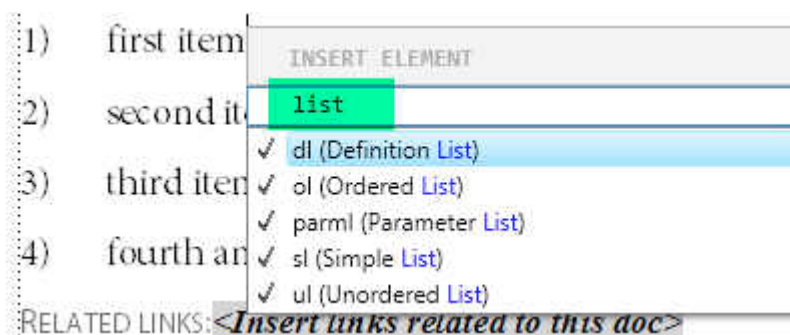
Get back your previous session with just one click. Automatically restore files that were open in the last session, pods and panels in use as well as their location, and the selected authoring view. Note: this is a huge time-saver for localization and LSP staff who must frequently work with up to one dozen documents at a time. Now, if they need to shut down their system for a break or end-of-day, upon restarting FrameMaker, staff may pick up with where they left off: with the same documents on the same pages they last viewed.

Regular expression support in Find/Change

Leverage regular expressions to carry out complex searches, or 'find and replace' on content in WYSIWYG and Author views. Use any of the three regular expression search engines—Perl, Grep, Egrep—and easily search body text, footnotes, markers, variables, and more. Regular expression search uses the Boost library, with Perl as the default search engine. Note that this can be particularly useful for reversing the order of components in calendar dates or in the display of currency.

Rapid authoring

Speed up XML authoring with a range of productivity features, such as Intelliprompt of elements and attributes, template-based structure suggestions, automatic insertion of end tags, dynamic validation, and pretty printing. Work faster with large XML/DITA files and DITA maps in multiauthoring environments. In this screen capture, the author is using a smart catalog keyboard shortcut to insert an element, and has typed in "list" to remove all elements that are not associated with that structure.



XSLT 2.0 support

Create a variety of transformation scenarios by applying predefined or customized XSLT to XML files. Execute these transformations using a range of JAXP-compliant processors, such as Xalan and Saxon. Define the scope of the transformation—a file, all files in a folder, DITA map, or all children of a book.

XPath 2.0 support

Eliminate the need to browse through numerous lines of XML code. Use the XPath Pod to find your elements and automatically apply functions to them. Customize your query builder, get suggestions, define the scope of queries, use the history drop-down feature, and save search results for greater productivity.

Filtering by attribute

Enable better management and reuse of content by taking advantage of support for multivalued attributes and the ability to import expressions from one document to another.

Attribute editor

Define and edit attributes for declarations in markup to perform multiple functions: record the status of content within elements, allow cross-referencing between elements, or specify how to format an element.

FrameMaker Conversion Tables

Quickly and easily convert your legacy content to structured content using the FrameMaker Conversion Tables. Generate a table of styles which can be mapped to structure elements. The mappings are stored as conversion tables which can be used to convert any unstructured document to structured.

Simultaneous PDF-based reviews and authoring

Work more productively by conducting authoring and reviewing in parallel. Edit the source content without waiting for all review comments to come in. Review comments are imported correctly in the modified document as long as the commented content has not changed in the source document.

Out-of-the-box integration with Adobe Experience Manager

Manage content effectively using new native integration with Adobe Experience Manager, formerly known as Adobe CQ. Check out, check in, upload, copy, move, and create files. Get digital asset management, metadata support, see version history, show dependents, and more. Customize Adobe Experience Manager by changing preferences.

Out-of-the-box support for MathML equations

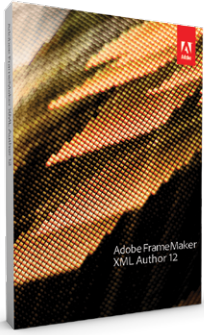
Create or edit MathML equations within FrameMaker 12 thanks to native integration with MathFlow™ Structure and Style editors from Design Science. Import a MathML equation into FrameMaker, as an image, auto-size it, and save its source in the XML file. Render all documents including XML, with embedded MathML equations.

Publish anytime, anywhere, to any device

Easily publish content to multiple channels, formats, and screens natively from within FrameMaker 12. Generate EPUB 3, KF8, MOBI, WebHelp, CHM, and HTML5 output at no extra cost, without any plug-ins.

New product: FrameMaker XML Author 12

This new product is a less expensive version of FrameMaker that is designed to be a standalone, end-to-end XML publishing solution. It is fully compliant with the DITA Open Toolkit to achieve publishing output. Unlike full FrameMaker, this product does not provide tools for developing structured templates or structured applications. But it is fully DITA 1.2 compliant out-of-the-box, at a substantially reduced price level.



Adobe FrameMaker XML Author 12

To find out more about and try Adobe FrameMaker XML Author 12, please visit the website now at www.adobe.com/products/framemakerxmlauthor.

In conclusion

Although conversion to XML in and of itself will eventually lead to time and cost savings in translation, the savings can be more dramatic when FrameMaker is employed as a solution. In short, FrameMaker accommodates for the "human factor" of staff and key stakeholders who may have minimal time to come up to speed on a new solution.

In today's demanding workplace, we are faced with more types of content to publish in more combinations for delivery to more platforms. With that challenge, FrameMaker will be highly appreciated by your team for "hiding" unnecessary code and complexity when you just need to concentrate on creating good, solid content.

About the author:



Maxwell Hoffmann is Adobe's product evangelist for Technical Communications and is based out of Portland, Oregon. He has over 20 years of experience with both FrameMaker and technical publishing (across print and online Help projects). Hoffmann spent over 13 years in the language translation industry managing multilingual production on high page count projects for aerospace, government, healthcare and medical devices.

