Text Structure Consulting, Inc.

17225 San Franciscan Dr.
Castro Valley, CA 94552
voice/fax: (510) 583-1505
http://www.txstruct.com

Lynne A. Price, President

September 25, 2012

FrameMaker 11—Adobe Listens

Max Hoffmann, Adobe's product evangelist for technical publishing, describes FrameMaker 11 as "the biggest release of FrameMaker in 10 years." No question that this new release includes very significant enhancements. The level of Adobe's commitment to the product is further indicated by the smaller changes and fixes to outstanding problems that accompany the major new capabilities. The pre-release period was longer than those for other recent releases. As a result, instead of merely identifying bugs, testers could make suggestions about both new and long-standing functionality, some of which were incorporated into the final product. FrameMaker 11 is not perfect—like all software in active use, it suggests to users how it can be improved.

Systematic descriptions of FrameMaker 11 are readily available. This review instead describes one user's reaction to some of the new features and the situations in which they may be used.

XML View and XSLT

Perhaps the most significant FrameMaker 11 enhancement, at least for structured documents, is the new XML View with the XSLT and XPath support it offers. More complete descriptions are available elsewhere (see, for example, Tassos Anastasiou's comments). In brief, XML View provides a text editor for XML documents, with optional pretty-printing and a tree view that is similar to the Structure View in the original structured FrameMaker environment (now called WYSIWYG View). There's also an Author View which displays XML documents in a structured format similar to that of WYSIWYG View but does not divide the document into pages or provide access to formatting commands.

In addition to editing XML text, XML View allows XSLT transformations (XSLT 1.0 or XSLT 2.0) to be run on individual documents or various collections of documents. It also supports using XPath expressions to search for particular constructs within a single XML document, all open XML documents, a FrameMaker book, a DITA map, or a folder.

Transforming multiple files with a single command and using XSLT 2.0 were not possible in earlier versions of FrameMaker. Users could, of course, switch to an external XSLT engine to run transformations and then open the transformed documents within FrameMaker. I was surprised at how much more convenient the new FrameMaker 11 capabilities are. Several details contribute to this ease of use:

1. XML documents frequently use relative filenames to identify schemas or DTDs and other external entities. To process such documents, software must be able to find the referenced files. (The necessary mappings can be specified with an OASIS XML catalog or some other means. FrameMaker partially supports XML catalogs, and also allows the equivalent

information to be specified directly in an XML application.) If documents are to be processed with FrameMaker as well as another tool, the mappings between relative and absolute filenames must be specified for both tools. Even if both tools use a single XML catalog, the catalog must be installed or identified in both. FrameMaker 11, however, uses the same entity mappings for opening and formatting an XML document as it does for applying XSLT. The user must provide the information, but only once.

2. FrameMaker 11 allows XSLT transformations to be associated with an XML application. Such transformations can then be applied to all documents in a folder that it automatically associates with the application through their root elements or the public identifier of their DTDs. For instance, suppose an organization maintains several books, each stored in a subfolder of a primary folder. Each book consists of multiple XML documents, with the possible root elements `front-matter`, `chapter`, `appendix`, `glossary`, and `references`. To modify the structure of all the `glossary` documents but none of the others, the user need only apply the appropriate XSLT transformation to the primary folder.

3. In FrameMaker 11, switching between display of an XML source document and its formatted version is trivial. The single click or keyboard shortcut needed is much easier than working through the Reading Text File dialog box that was necessary in previous versions to see the XML source. It's also easier than using different software to look at the XML file.

XML View can be improved (for example, it would be great to apply XSLT to a structured document even if it hasn't been saved as XML), but FrameMaker 11's version is a significant productivity aid. Besides, we users can hope this capability will be extended in the future!

Evolution of guided editing

Since structure was added to FrameMaker almost twenty years ago, the Structure View and Element Catalog have guided authors in creating valid documents. The Structure View indicates when elements (and attributes) are missing or erroneous, and the Element Catalog lists the elements that are permitted at the current location. Consider something as straightforward as the following example:
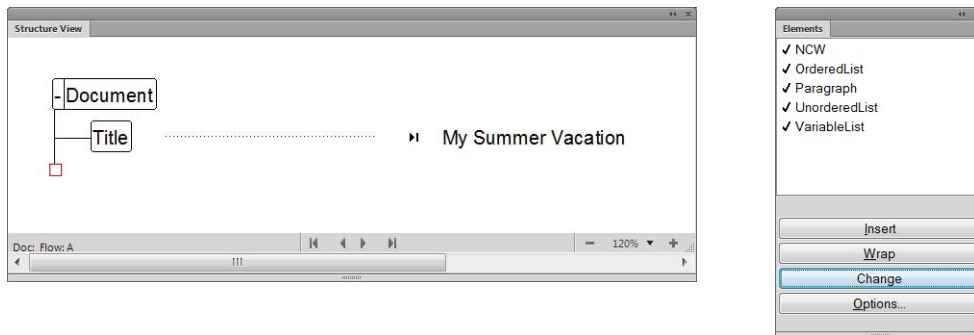
Here, the red "hole" in the Structure View indicates that more content is required within the `Document` element and the Element Catalog lists the elements that are allowed after the `Title`. This helpful information does not indicate whether one of the listed elements is sufficient to make the `Document` valid, nor does it explain the rather obtuse `NCW` element name. Authors need training and documentation about the tag set they are using to understand such details.
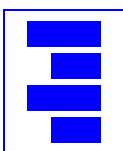
FrameMaker 10 took a step toward minimizing how much authors need study such documentation with the introduction of **descriptive tags**, annotations in the Element Catalog that provide a brief description of each available element. Adding descriptive tags to the above example yields:
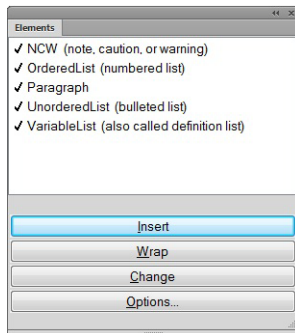
which explains that NCW is used for notes, cautions, and warnings; reminds authors that the term "ordered list" refers to numbered lists, while "unordered list" refers to bulleted lists; and explains that the term "variable list" refers to a construct sometimes described as a "definition list".

FrameMaker 11 provides another form of context-sensitive, as-needed documentation with the introduction of **banner text**. Banner text is temporary text, defined in an EDD, that appears within an empty container. It disappears as soon as the author inserts actual content into the element. Banner text calls attention to incomplete elements, but it is also an opportunity to remind authors of how new elements should be used. Here, for example, the NCW element by default is used for notes, but its banner text explains how to create a caution or warning:

> **Note:** *For a caution or warning, set the Type attribute of the NCW element.*

Object catalog

A long-time enhancement request has been for the ability to set user preferences for properties applied to new images, anchored frames, and other objects. FrameMaker 11 adds an object style catalog and Designer which allows users to apply saved properties to new or existing objects. The object style catalog is stored in a document or template, so the user never needs to see lists of object styles that are not relevant to the current document.

Workspace enhancements

FrameMaker 9 introduced arrangements of pods and panels called **workspaces**. A user can create and modify workspaces to save the various configurations he prefers for editing different types of documents or performing different tasks. FrameMaker 11 adds menus and toolbars to workspaces. With this ability, organizations as well as individual users may wish to define workspaces. For example, an organization may provide a workspace with abbreviated menus for authors and more complete menus for developers who maintain templates. It's always been possible for authors and developers to use different menus, but workspaces make it easier for developers to switch between the two models. Furthermore, workspaces make it easier for a user to move between his normal work environment and FrameMaker's default configuration, making it more likely that users will take advantage of FrameMaker's features for more types of documents.
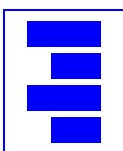
Correction of oversights

As mentioned above, FrameMaker 11 corrects some oversights in earlier versions. For example, custom ruling and shading of table cells can now be round-tripped between FrameMaker and XML without a custom API client.

A change that affects more users is the method of tagging various constructs without using a mouse. FrameMaker allows users to insert, wrap, or change elements, tag paragraphs or selected characters, apply or remove condition tags, and insert variables from the keyboard. This ability is convenient whether or not the possible tags are displayed on screen because the user can work quickly without constantly reaching for a list of tags and then reaching back to the document window. Before FrameMaker 11, these operations were done with **quick keys**. Quick keys displayed possible tags in the desired category in the lower-left corner of the FrameMaker window. Using the up or down arrow keys or by typing the first characters of the desired tag, the user displays the desired tag and then presses the Enter key to apply it. The following example shows the FrameMaker 10 display when the user is about to apply paragraph tag `Heading1` to the current paragraph.
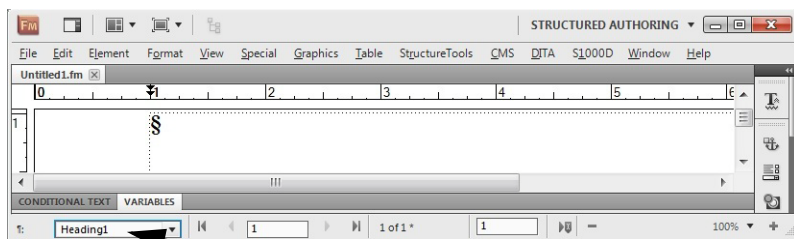
the selected tag

indicates display of paragraph tags

A frequent complaint since the user interface was revamped in FrameMaker 9 was the unreliability of quick keys. FrameMaker 11 did not merely restore the earlier quick key behavior. Instead, it replaced quick keys with **smart insert**. This feature is useful but poorly named—as well as inserting variables and elements, its uses include operations such as removing condition tags and changing elements that do not insert anything new into the document. Smart insert replaces the quick key area in the lower left with a list of available tags that appears near the current location in the document:
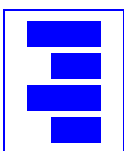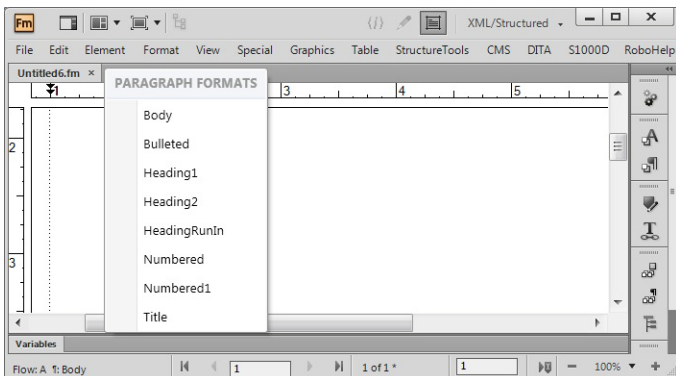
Not only does smart insert show the possible tags in a list instead of one at a time, but the location of the list moves with the location of the current position on the screen. Quick keys allowed the user to leave his hands on the keyboard in order to select a tag; smart insert also allows him to apply tags without moving his glance significantly from the area he is editing.

Customizability

One of FrameMaker's strengths is the flexibility provided by its many configurable features. Configurable new features in FrameMaker 11 include:

1. Users have a choice of the XSLT engine used when running XSLT in XML View. Saxon 9.4 and Xalan 2.0 are included with FrameMaker 11, but users can use any JAXP-compliant XSLT processor.

   Note: FrameMaker optionally runs XSLT when opening or saving an XML document. The XSLT processor used for such pre- and post-transformations is still Xalan, which does not support XSLT 2.0.

2. When a FrameMaker 11 user pastes material copied from HTML, Microsoft® Word, Microsoft® Excel, and Microsoft® Outlook into a structured document, FrameMaker creates an appropriate hierarchy of elements. This powerful new feature, called **smart paste**, is configurable. The user provides an XSLT transform to generate the desired structure.

3. Many character properties of banner text are configurable. Users can also indicate whether banner text is displayed whenever a container is empty or only when an empty container is first inserted. Unfortunately, these settings cannot be changed for individual documents or individual elements within a single document.

4. The display in XML View is configurable. Users can specify the font family used, colors for different parts of an XML document (attributes, comments, processing instructions, and so forth), whether or not to display line numbers, and whether to show long lines in a file over multiple lines on the display.

Conclusions

As a consultant specializing in structured FrameMaker, I expect to use XML View extensively. End users are less likely to do so, although I have recommended that some structured documents be converted to XML solely for the ability of using XPath. Many of my customers can benefit from the new workspace features, banner text and descriptive tags, smart insert, smart paste, and object style catalogs. I hope FrameMaker 11 sets a trend for further enhancement of the FrameMaker product line.