

Creating XML Content

tools of the trade
XML HTML

ments with other organizations, or even meet contract or industry requirements that documents be delivered in XML. Migrating to XML publishing warrants a careful evaluation of your organization's requirements, benefits of meeting those requirements, and costs of moving to XML publishing (including design time, tools, and training costs).

"How will our authors create XML content?" is one of many questions an organization must answer when migrating to XML publishing. When evaluating XML authoring tools for content development, you will need to consider how each meets your organization's requirements. This article presents factors in choosing XML authoring tools and briefly reviews three popular commercially available XML authoring tools: ArborText *Epic Editor*, Adobe *FrameMaker 7*, and SoftQuad's *XMetaL*.

XML-based publishing can provide many benefits.

These include the ability to more easily manage and reuse content, publish content in multiple formats from a single source, build customized documents for particular users or products, share docu-

BY ALAN R. HOUSER
Senior Member
Pittsburgh Chapter



XML Editors: Orient Yourself

Perhaps the most important consideration when choosing an XML authoring tool is to be sure that it is appropriate for creating technical documents. The strengths of XML—embedded metadata, platform independence, and relatively simple syntax—make it a great choice not only for technical communicators but also for software developers who need to create programs for exchanging and managing information.

Many tools are available to help software developers work with XML. However, XML editors that are popular with software developers will probably be disappointing to technical communicators. These XML editors tend to present a user interface that emphasizes the *structure* of an XML document instead of the content. (For an example of a developer-oriented XML editor, see Figure 1.) These tools are not appropriate for content creators for several reasons:

Lack of formatting feedback. Because XML content tends to be published to multiple output formats and may be combined in different ways to create customized documents, XML authoring tools are typically *not* WYSIWYG. However, many writers prefer some sort of formatted screen rendition, even if it does not exactly match the final output format. Content developers typically do not need to see the exact final format of the headings as they are rendered in print, in a Web browser, or in some other display application. They typically do prefer, however, that the heading receive some sort of special formatting—perhaps set off with a larger font, bolded font face, and extra vertical space—while they are authoring a document.

If you think screen formatting is not important to you, imagine working in an editor that does not wrap long lines. Many developer-oriented XML editors do not.

Lack of content-friendly features. Do you use a spell-checker? Online thesaurus? How about a cross-reference tool? If you use any of these features, you may be disappointed and frustrated by developer-oriented XML editors that do not provide them.

Lack of important document components. You may be surprised to find that your developer-oriented XML editor does not provide an interface for creating tables or inserting graphics. Technically, a table is a presentation format, but it is, nonetheless, an important tool for technical communicators. Document-oriented XML authoring tools support tables, graphics, index entries, cross-references, and other important document components.

After you have identified XML authoring tools that are appropriate for creating content, the following are other important considerations:

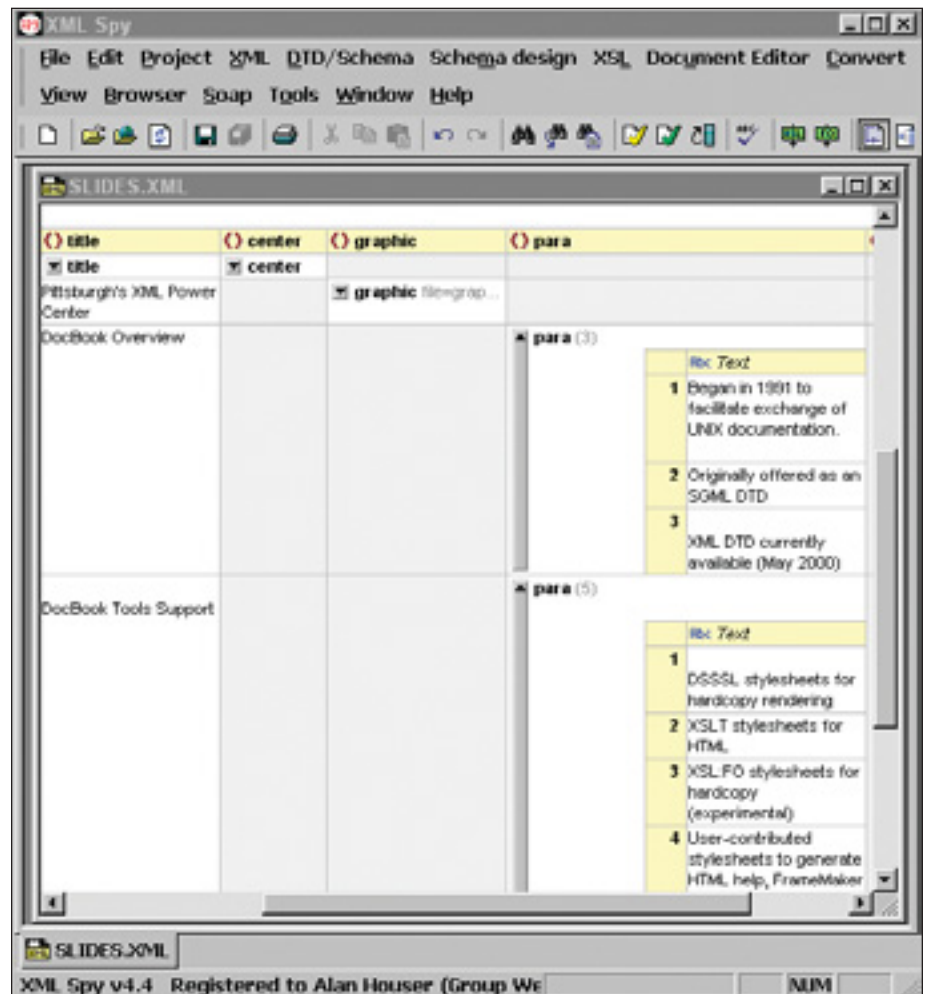
Conformance to your standards. XML standards are new and evolving. The XML-related standards that are important to your organization might differ from those that are important to others. Do you need to validate your documents against an XML

Schema? Do you need to include scalable vector graphics in your documents? If so, be sure your XML authoring tool supports these standards. (See the sidebar at the end of this article for information about XML Schema and scalable vector graphics).

Integration with your IT environment. Your organization may use a content management system (CMS) for revision control and tracking of XML content. Verify whether the XML authoring tools that you consider will integrate with your CMS. Ideally, your XML authoring tool will provide menu options for checking files in and out of the CMS.

Support for conventional publishing. XML is a great data format but a poor publishing format. To publish XML documents, you must first convert the XML into a format that the display device can understand.

Figure 1. Screen interface of a popular developer-oriented XML editor.



The most popular display formats include HTML for Web browsers, PDF for print or richly formatted on-screen viewing, wireless markup language (WML) for display on small screens, Rich Text Format (RTF), and PostScript.

If your authoring tool does not convert XML to these formats, you will need to buy (or develop, possibly with the help of an XML expert) a tool to do this.

Before You Begin

You might think that you can buy an XML authoring tool and get started right away. However, you must also set up the support files that your authoring tool requires for editing valid XML documents. These support files do the following:

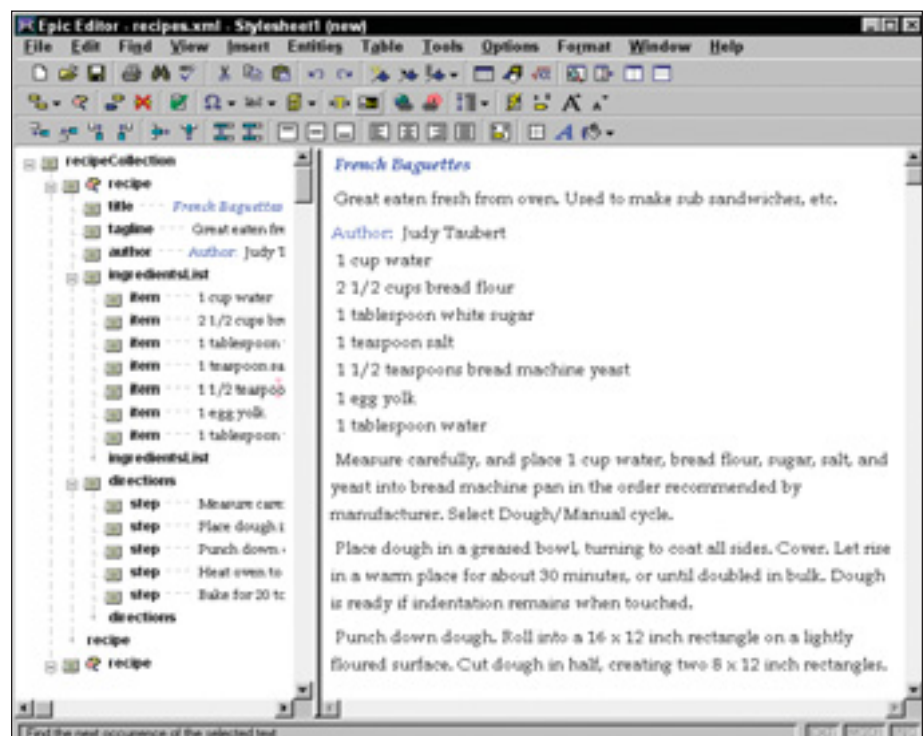
- Specify the Document Type Definition (DTD) that your organization has chosen or created, to guide authors in creating valid XML content.
- Specify the screen display format (and possibly print rendering) of the XML documents.
- Customize (optionally) the authoring interface for a particular type of document.
- Specify how XML elements will be translated into “document” elements. For example, which XML elements represent graphics? Which represent tables? Which represent block content (like body paragraphs)? Which represent inline content (like part numbers)?

Setting up an XML authoring tool can be a complex task. Fortunately, the XML authoring tools reviewed here include set-up files for one or more popular XML DTDs. You can modify these configurations for your own environment, or you can use them as models for configurations that you create from scratch. If you are using an industry-standard DTD, verify that the authoring tools you evaluate provide support files for that DTD.

Content-Oriented XML Tools

Of the commercially available XML authoring tools, three are key players in the market. ArborText’s *Epic Editor*, Adobe’s *FrameMaker 7*, and *XMetaL* by SoftQuad (now owned by Corel Corporation) share the following characteristics:

Figure 2. Epic Editor interface, with document window and document map.



- Each tool has a history that spans several years, and each was originally developed as a tool for creating SGML (the predecessor of XML) documents.
- Each tool supports authors in creating XML documents without the need to know XML syntax.
- Each tool provides full control over an XML document. Specifically, these tools allow authors to create content that conforms to any DTD and to edit both XML elements and attributes. Some XML authoring tools support only a single DTD or provide a forms-based interface that only allows authors to create element content.

Each of these tools is a capable choice for creating XML content. All provide the following features:

- Word processor interface, with word processing features, such as drag-and-drop editing, copy and paste, spell-checking, and shortcut toolbars
- Optional view that shows XML tags within the document
- Concurrent view that shows a graphical representation of the document structure and hierarchy
- Feedback to the writer regarding legal

- tags at any given cursor location
- Continuous validation of an XML document against a DTD or XML Schema
- Ability to create and edit both XML element content and attribute content
- Ability to import documents created by other applications (such as Microsoft Word)
- Ability to create tables using a built-in table editor

Epic Editor

Arbortext’s *Epic Editor* has traditionally been targeted to authoring teams that create large amounts of XML content. (See Figure 2 to view *Epic Editor*’s interface.) *Epic Editor* shines in its capability to manage large document sets comprising hundreds or thousands of source files.

Epic Editor has long supported both XML and SGML versions of the DocBook DTD. The DocBook DTD was originally designed as an SGML DTD to support collaboration among several companies that contributed to a set of computer software documentation. DocBook has evolved into a large XML vocabulary (with nearly 400 element names) for creating technical documentation. (For more information about DocBook, see www.oasis-open.org/docbook.)

Epic Editor is highly customizable, and it supports creation of forms-based interfaces for authors. For example, if your writers create product data sheets, they will typically enter one of a finite set of units. You can minimize your writers' effort and reduce errors by presenting the possible units in a pop-up menu instead of having them type the units each time.

Of the three XML authoring tools presented here, *Epic Editor* provides the largest number of industry-standard DTDs—DocBook, XHTML, Telecommunications Interchange (TCIF/TIM), and Military spec 38784C (CALs). If your organization is using one of these DTDs, the preconfigured support files will allow you to quickly set up your authoring environment.

Other distinctive *Epic Editor* features include the following:

- Change tracking
- Handling of multi-level, compound documents
- Multiple levels of undo
- GUI-based style sheet editor
- Can be customized using C, C++, Java, Javascript, Visual Basic, TCL, Perl, and Python
- Built-in scripting language (ACL)

Epic Editor is available for \$695 USD.

FrameMaker 7

Before releasing *FrameMaker 7* in May 2002, Adobe offered two versions of its technical authoring and publishing tool—standard *FrameMaker* and *FrameMaker+SGML*. *FrameMaker* had traditionally been a tool for creating and publishing technical documents. *FrameMaker* has long been popular with the technical publishing community because of features that include robust handling of long documents, ability to create complex tables, and reliable auto-numbering. *FrameMaker+SGML* combined the features of standard *FrameMaker* with the capability to import and export SGML. *FrameMaker+SGML* was widely used in several niche markets in which SGML is important, such as the airline and manufacturing industries.

Later versions of *FrameMaker+SGML* supported saving SGML documents with

XML syntax. However, *FrameMaker+SGML* did not support opening and editing XML documents. *FrameMaker 7* replaces previous versions of both *FrameMaker* and *FrameMaker+SGML*. *FrameMaker 7* includes all the capabilities of the former products, and adds the capability to import, create, edit, and save XML documents. (Figure 3 shows a view of the *FrameMaker 7* interface.)

FrameMaker 7's document display is based

on a page metaphor. Even if you are creating XML content that won't be printed, *FrameMaker 7*'s display template includes page height and width information, margins, and number of columns. This does not affect the XML documents created in *FrameMaker 7*; it simply means that your authors will work in a view that closely resembles conventional page-oriented word processing. It also means that printing XML documents (or creating PDF

Figure 3. *FrameMaker 7* interface, with document window and structure view.

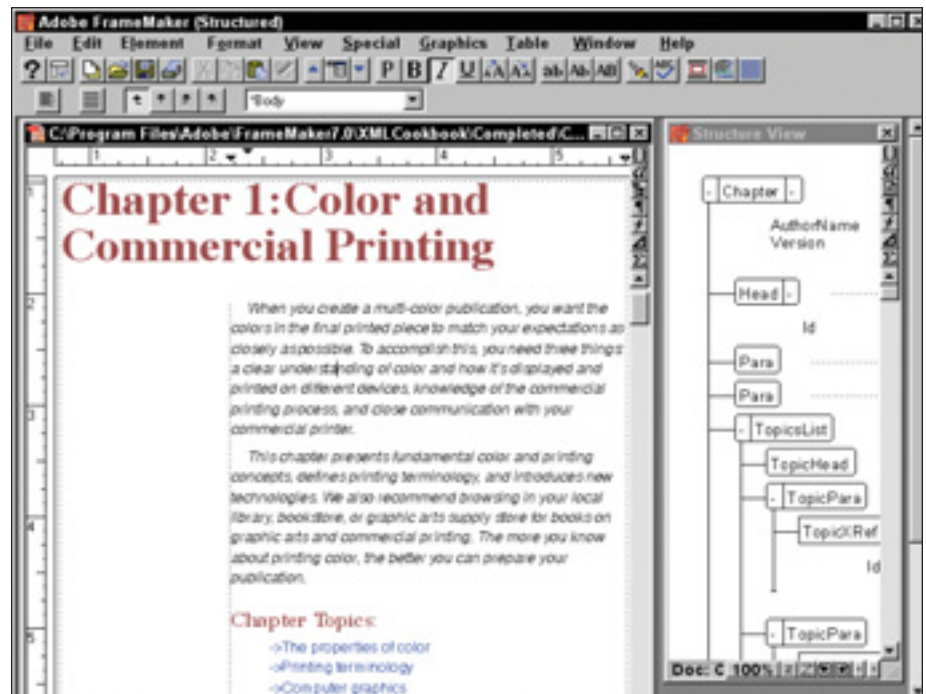
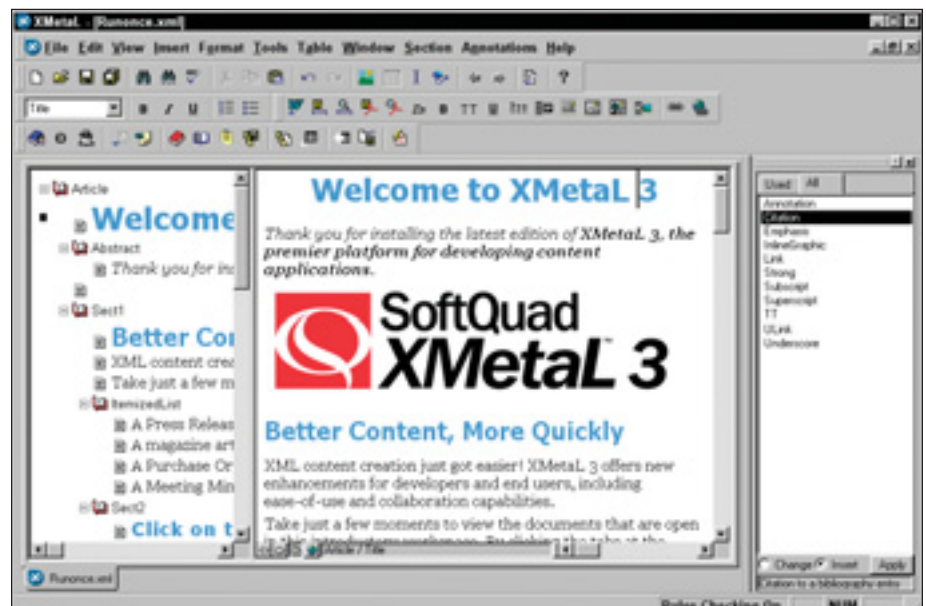


Figure 4. *XMetaL* interface, with structure view, document window, and list of elements.



files, with *FrameMaker 7*'s integrated Acrobat Distiller application) is very simple.

Other distinctive *FrameMaker 7* features include the following:

- WYSIWYG view for printing
- Built-in conversion of non-XML documents to XML
- Support for non-XML document authoring (in conventional *FrameMaker* format)

FrameMaker 7 is available for \$799 USD.

XML

If *FrameMaker 7* is particularly well suited for creating XML documents that will be printed, *XMetaL* may be particularly well suited for authoring environments in which conventional printing is not important. The *XMetaL* feature set is geared for creating content for online display, or content that will be printed separately from the authoring application.

Unlike *Epic Editor* and *FrameMaker 7*, *XMetaL* supports validating document content against an XML Schema. (Note: Arbortext plans to release a version of *Epic Editor* with XML Schema support before this article goes to press.) XML Schema is a recent Recommendation of the W3C that expands upon the capabilities of XML DTDs. A key benefit of XML Schema for content creators is stronger element type checking. An XML authoring tool that supports XML Schema can, for example, verify that a *price* element contains a number, that a *part_number* element contains a legal part number for your company's products, or that a *unit* element is a legal unit of measure.

Unlike *Epic Editor* and *FrameMaker 7*, *XMetaL* does not provide "out of the box" support for the DocBook DTD. *XMetaL* includes configuration files for XHTML and several example DTDs. Like *Epic Editor*, *XMetaL* provides the capability to create forms-based interfaces to assist writers in entering standard or frequently repeated information.

XMetaL provides perhaps the smallest amount of functionality of the three XML content creation tools, especially in the area of managing large sets of documents (like *Epic Editor*) or print support (like *FrameMaker*). However, *XMetaL* pro-

vides a highly customizable, yet relatively simple interface, and it can be a great choice for an authoring tool that can be customized to meet the needs of content creators. (For a view of *XMetaL*'s interface, see Figure 4.)

Other distinctive *XMetaL* features include the following:

- Customizable in several different programming languages
- Macro recorder for automating repetitive tasks
- Built-in conversion of Microsoft *Word* documents to XML
- Change tracking, including ability to accept or reject changes to a document

XMetaL is available for \$495 USD.

Conclusion


XML has become a popular file format for both software developers and content developers. While many XML editors are available, not all meet the particular needs of content developers. Of the document-oriented XML authoring tools, *Epic Editor*, *FrameMaker 7*, and *XMetaL* share many common features that are useful and necessary for content developers, while each product presents its own unique strengths.

Contact Information

Each of these XML authoring tools is available in an evaluation version, with extensive documentation to guide evaluators. If you are considering an XML content creation tool, contact the manufacturer for an evaluation version.

Epic Editor
Arbortext, Inc.
www.arbortext.com

FrameMaker 7
Adobe Systems Inc.
www.adobe.com

XMetaL
Software, Ltd.
www.softquad.com 

Alan Houser is a principal partner of Group Wellesley, a Pittsburgh, Pennsylvania-based company that specializes in information management, XML consulting and training, and technical writing. He is president of the Pittsburgh Chapter STC and is co-author of XML Weekend Crash Course, published by John Wiley and Sons. You can reach him via e-mail to arh@groupwellesley.com.

XML

and Related Specifications for Publishing

Technical communicators may find the following Recommendations of the World Wide Web Consortium (W3C) particularly interesting. For more information about each, see the W3C Web site at www.w3.org.

Extensible Stylesheet Language—Formatting Objects (XSL-FO). A vocabulary for rendering XML documents on a printed page.

Extensible Stylesheet Language—Transformations (XSLT): A specification for transforming and manipulating content from XML documents. Typically used for transforming XML documents from an authoring vocabulary (such as DocBook) to a presentation vocabulary (such as XHTML or XSL-FO).

Scalable Vector Graphics (SVG): A file format for graphics that is expressed in XML syntax.

XHTML: A reformulation of HTML that conforms to XML syntax.

XML Schema: An alternative to DTDs for validating XML documents. In addition to validating the vocabulary and structure of an XML document, XML Schema allows an application to verify the data type (such as number, date, or string) of XML element content.

XML: A specification for combining both document content and information about the content (metadata) in a platform-independent file format. The XML Recommendation also defines the XML DTD, which specifies the legal vocabulary and structure of an XML document.