

Using XML

**A look at XML as a document
authoring and management tool**

Scott Nesbitt

-
- XMLmind XXE
 - oXygen
 - EpcEdit
 - XEmacs, with the psgml or psgmlx packages

Many of these applications enable you to create and edit documents in familiar word processor-like environment. Others are text-based applications with specific XML authoring features. Both types of applications have their strengths and weaknesses. The choice of tool you make will depend on how comfortable and efficient you and the members of your team are with using WYSIWYG or plain text tools.

Document Management Software

Some of the better XML-capable document management applications include:

- ArborText *Epic*
- XyEnterprise *Content@XML*
- Documentum *4i*
- Lightspeed *Astoria*
- Author-IT
- Live Linx
- Borges
- SiberSafe

Not surprisingly, these applications have similar core functions – collaboration, re-use of information, and version control. Others have specialized features like the ability to map out and track workflows, as well as electronic review.

Most of these programs also integrate with popular authoring tools like *FrameMaker* and *ADEPT*Editor* (some handle Microsoft *Word* files as well), and can generate output in several formats. Most are also fairly expensive, but you get what you pay for. In most cases, you are getting quite a lot.

Of course, you're not limited to expensive commercial offerings. Some shops use change tracking and version control tools like CVS and Perforce. If you are interested in lower-cost alternatives like these, visit SourceForge and do a search for content management.

Choosing a tool isn't as cut and dry as it may seem. You have to decide what your needs are, then investigate as many applications as you can. You also have to take into account the authoring tool you are currently using. If it is not supported by the XML system you have chosen, there will be an additional expense in purchasing and learning a supported application.

Chances are, you won't be creating documentation from scratch; you will have an existing documentation set. If you are in the position of starting from scratch, you can start authoring in XML. For existing documentation, *FrameMaker* is the most widely-supported application, although certain tools only integrate with *Word* for Windows.

That's Nice, But ...

... what does all of this mean to you? Simply put, the tree-like structure of an XML document allows you to easily manage your information. And you can chunk the information to quickly extract what need when you need it. Not only that, you can take the information you need from a document and use it elsewhere. For example, if you are building an online Help system, you can extract all the relevant procedures but leave out any superfluous overview information. You can do this using, say, the conditional text feature in *FrameMaker* but doing so can be awkward and time consuming. On top of that, using XML enables you to re-use standard chapters – introductions, glossaries, “how to use this program”-type sections, etc. – quickly and easily.

What Now?

You have the content. You have the tools. Now what do you do? One of the first things is to create a functional specification². The functional specification will literally maps out what your manuals will contain and how you intend to structure the information in the manuals. And that leads to the next step ...

... analyzing your documents

Your analysis will help you:

1. Decide on a sufficient level of granularity in the documents. You must decide how much granularity is enough, based on your present and future needs
2. Determine how to order the elements in the documentation. You must consider *every* way in which elements nest

²See chapter 7 of *Mastering XML* by Linda Burman, Chuck White, and Anne Navarro for more information.

Note: If you need more information on CSS, you can find it a numerous Web sites including Web Developer's Virtual Library and WebReference.

Conclusion

XML can change the way in which you work with documentation. And not only in the way the documentation is created and managed. By moving to XML, there will be a fundamental shift in the way you view documentation. You will move from seeing manuals and Help files as documents with a beginning, a middle, and an end (perhaps a holdover from the English/Arts backgrounds of many writers), to seeing documentation as structured components of information. You can then take these components and fit them together as needed, as if they were blocks of Lego.

Getting to the point where you can use XML effectively requires a lot of work – from learning the language to choosing an XML-enabled tool to analyzing documents to building DTDs. In the end, though, all the effort expended will be worthwhile. Moving to XML might not improve the quality of your documentation, but it can streamline the process of creating and managing documents.

Appendix 1: A Manual Chapter Coded in XML

This chapter from a manual for a fictitious XML editor illustrates one way to tag a document.

```
<chapter>
  <heading>
    <chapter.title>Using Your XML Editor</chapter.title>
  </heading>
  <chapter.body>
    <title>Using Your XML Editor</title>

    <section.overview>
      <section.title>Introduction</section.title>
      <para>Your XML editor is one of your most important tools.</para>
    </section.overview>

    <section>
      <section.title>What the Editor Can Do for You</section.title>
      <para>This XML editor has many features to assist you, including:</para>
      <bullet.list>
        <bulletlist.item>Easy editing in a standard environment</bulletlist.item>
```


