



# BRIEFING FOR MANAGERS

BY ADAM K. WATTS  
SPECIAL TO GOVERNMENT  
TECHNOLOGY

*How does XML differ from HTML? Is it something new that will sweep away all the hard work on your portal, or can XML and HTML co-exist?*

WITH RAPID CHANGES AND DEVELOPMENTS IN WEB-BASED technologies, it can be difficult for government IT staff to determine which route to take for the most effective method of information presentation. As the trend continues toward "thin clients" and Web-enabled interfaces, more and more IT departments are looking at XML. How does it differ from standard HTML? How difficult is it to implement? What benefits would it bring?

Like HTML, XML (EXtensible Markup Language) is a branch of the SGML (Standard Generalized Markup Language) family tree. Although its original purpose focused on large-scale electronic publishing, XML is now recognized for its ability to exchange a wide variety of information over the Internet. Thus it seems a natural partner for electronic government applications.

## WILL XML REPLACE HTML?

One misconception is that XML will replace HTML. The confusion likely arises from the fact that, like HTML, XML is for online use, its documents are viewed by Web browsers, and the "plain English" coding of XML appears to be very similar to the text and "tags" of HTML. For example, HTML tags are commands, identified by angle brackets, that tell the browser how to display information. A `<b>` indicates the start of a portion of text to be bolded, and a `</b>` indicates the end of the portion of text to be bolded.

XML addresses the problem of online data retrieval and access. HTML handles presentation of information. HTML says: Put a new paragraph there. Put a heading here. Center this picture there.

XML says: Access this database; locate that type of information; output the results like this.

Though XML cannot replace HTML, it will certainly replace some of the uses to which HTML has been applied. The simple fact is that most Web pages are static. That is, the content of the pages does not change on a regular or frequent basis. HTML still remains the best tool for static pages.

XML is not a newer version of HTML. Choosing one or the other is like comparing apples and oranges. XML will provide Web developers with a simpler way to generate Web pages on-the-fly. Such pages can have different content every time they are accessed. Content changes can originate from how the user requests information or from updates to the database from which the XML file obtains the information being presented.

## EXTENSIBILITY: THE DOUBLE-EDGED BLADE

The term "extensible" simply means that the set of commands being used can be expanded as the need arises. With HTML, all commands are predefined. A Web developer is forced to operate within the boundaries set by the existing coding commands.

One advantage offered by XML is that, instead of using predefined commands, programmers can create their own commands to accomplish their ends. That advantage, however, can also be a disadvantage. One need only harken back to the days when every computer company wrote software in its own proprietary language to realize the inherent dangers of allowing programmers to make up their own languages as they go, lest interoperability become a thing of the past.

Extensibility, then, is a blade that cuts both ways. For two systems or programs to communicate, they need to understand and share the commands being used, either directly or through a third-party interface or translator.

On the surface, this may seem like a familiar problem and one might wonder, "Why complicate things further by adding XML into the picture?" However, XML does provide a simpler means of getting two disparate systems to communicate. If an agency is looking for a way to allow its staff or constituents to search and access information in a changing database via Internet or intranet, an XML strategy should definitely be investigated.

## PUTTING IT TOGETHER

Making use of XML itself may not require upgrading hardware or software. Most HTML Web servers are up to the task. To make database information available does require a reference table to be constructed for the database. This table can then be accessed and the information loaded into a viewable page.

One surprising hurdle to face with XML is its similarity to HTML. Those who are familiar with HTML will likely need to change their habits to work with XML. Though it has been recommended that HTML tags be typed in lowercase, it has never been required. XML is case sensitive, and therefore would view `<b>` and `<B>` as two different commands. Many HTML tags do not require a closing tag, or are very forgiving if one is forgotten. XML requires that every tag have a closing tag, such as `<b>` and `</b>`.

Additionally, in XML, all attributes must be enclosed in quotation marks. With HTML you could specify `width=100%`, but with XML it would need to be stated as `width="100%"` or the document could not be read. XML will never guess at what the designer really meant.

Fortunately, the World Wide Web Consortium has made verification utilities available for free. This makes it simple for Web designers to find and fix problems with the coding. These utilities, along with much more extensive information on XML, are available at [www.w3.org](http://www.w3.org).

## XML FOR JUSTICE COLLABORATION

Todd Pedwell, manager of the Justice Web Collaboratory at the Chicago-Kent College of Law, utilized XML technology to enhance Web site functionality. The site [www.judgelinek.org](http://www.judgelinek.org) is essentially a newsfeed service and online community for judges.

The project initially used two students three hours per day to search through different newsfeeds and revise HTML pages to make the new information available to members. Pedwell felt that these hours could be better spent.

A number of factors came into play to influence Pedwell's decision to implement XML. Many newsfeed services are making content available through XML. This meant that the infor-

**“UNDERSTAND WHAT YOU ARE GOING TO BUILD FIRST. UNDERSTAND YOUR NEEDS, HOW YOU ARE GOING TO USE XML AND WHAT PURPOSE IT WILL SERVE. THAT WILL TELL YOU IF XML IS YOUR ANSWER, AND IF IT IS, HOW YOU SHOULD IMPLEMENT IT.”**

— Todd Pedwell, manager, Justice Web Collaboratory

mation could be obtained in a preformatted state, requiring little or no data manipulation.

“We reviewed what the newsfeed services were doing,” Pedwell explained. “We found out what type of tags were common for the feeds that we needed and we built the site using those.”

Additionally, because students had a high interest in XML, that made it very easy to interest them in the project. This, combined with the fact that implementing XML would not require hardware or software upgrades, meant that the project could be done with a minimum of financial investment.

“We were able to do everything with the existing structure,” Pedwell explained. “We did end up doing more work on the SQL server than we originally thought, but that has given us even more flexibility than what we were initially trying to achieve.”

The Justice Web Collaboratory wanted to construct a judicial news service, where members could see current news-feeds pertaining to the judicial system, and also to make it possible for members to post news or announcements of their own.

“If the Federal Judges Association has an event or news they want to deliver, XML gives them the opportunity to add news to our newsfeed,” Pedwell said. “This goes a long way toward building a community.”

As a resolution to the problem, Pedwell's team built a database that generates an XML file that is loaded into a look-up table on the Web server. The news page can then pull the data from the look-up table whenever the page is accessed.

The first step was to integrate the table into the Web site. The next step involved creating the XML file. Initially, this was done manually. “Creating the XML file by hand wasn't very easy,” Pedwell explained. “XML is very strict, so we had a lot of errors.”

The next step was to create a database that would update the XML file hourly if there has been a change to the database. For the testing phase, an update to the XML file would also generate an e-mail notification. The team could then look over the file and, if it checked out fine, manually move it to the Web server.

Pedwell is ready for the final stage. He will move everything to the production server, eliminating the need for a manual file move.

“I'm really surprised at how smooth it went,” Pedwell remarked. “But we did a lot of research up front as far as interoperability with other systems and the tools available that we could use to help integration.”

According to Pedwell, up-front research is the key to an easy and successful implementation. “Understand what you are going to build first,” Pedwell recommended. “Understand your needs, how you are going to use XML and what purpose it will serve. That will tell you if XML is your answer, and if it is, how you should implement it.” ☉

*Adam K. Watts <techwriter@excite.com> is a freelance technical writer and editor based in Sacramento, Calif., whose projects included editing Web content for the energy conservation section of the MyCalifornia Web site.*