The Philadelphia Inquirer
PHILLY TECH
MAGAZINE

New Media RICHARD KOLENDA

XML: The Next Generation of Web Development

by Ric Kolenda

Okay, gang, I know you need another set of acronyms like you need a hole in your firewall, but this one just might change your life. Well, at least it will create a lot of work for you in the near future.

XML, or eXtensible Markup Language, is touted to eventually replace HTML as the language of the Web. This claim is hotly contested, but there is certainly enough buzz to take note of, if not believe, the hype. The problem with XML is that it's not quite as straightforward as HTML, so a lot of us non-coder types are having a hard time getting our arms around the concept. The simplest explanation is that XML is a set of standards for creating HTML-like tags and style sheets.

It might help to back up and look at the evolution of XML and HTML. Both are descendents of SGML, or Standard Generalized Markup Language. SGML was developed in the late 1970s as a standard way to organize structured documentation. But while HTML is a small, clearly defined subset of specific tags, XML conforms to the SGML standard but allows the user to define and interpret new, document-specific tags.

In short, HTML ignores tags the browser doesn't understand, but with XML, the browser can look-up and interpret those tags. Or as Norman Walsh says in his introduction for XML.com:

> *XML specifies neither semantics nor a tag set. In fact XML is really a meta-language for describing markup languages. In other words, XML provides a facility to define tags and the structural relationships between them.*

Others call XML "a metamarkup language" or "a metadata language." Any way you look at it, it's not your father's (or even your slightly older sister's) web language. But there is one major difference between HTML and XML, and pay attention here all you HTML hack artists: *unlike HTML where the browser will try to make sense of badly-written code, an XML document that is not "well-formed" will be simply ignored*. In other words it's right or it ain't showin' up. So the incentive is pretty high to create correct code, which actually makes the folks at Microsoft and Netscape very happy.

Browser Support

This is where things start to get tricky. Because XML is capable of interpreting its own tags, and because it is SGML-compliant, it should be theoretically browser-independent. That doesn't mean, however, that all browsers are created equal when interpreting XML applications. An XML *processor*, more commonly known as an XML *parser*, is necessary to convert XML into HTML. Microsoft began XML support in IE4, and has expanded it significantly in IE5, even creating proprietary add-ons. Much of the XML support involves Schemas (a kind of

limited DTD), Cascading Style Sheets (CSS) and Channel Definition Formats (CDF). Netscape uses an XSL they call Rich Site Summary (RSS), a set of tags for creating rich media channels. In any case, more and better parsers are available all the time which will make it easier than ever to create browser-independent yet highly formatted documents.

So what does all this mean to you? The ability to deliver structured data to the desktop will allow for client-side activities now confined to server-side middleware. This is especially true for things like e-commerce engines, search engines, and other data-processing intensive operations. So I recommend checking out some of the sites below to get your feet wet and see what's possible. You might even generate some XML, even an XSL or two. It's easier than you might think. Then tune in next time for some cool examples of XML implementations.

Short Glossary of XML Terms

Term	Definition
CSS	**Cascading Style Sheets** define style characteristics in a DOM. Cascading refers to the fact that styles can have hierarchies, with lower styles overriding those higher in the hierarchy.
DHTML	**Dynamic HypeText Markup Language** is basically HTML plus DOMs, CSSs and scripting languages like JavaScript and VBScript.
DOM	***Document Object Model*** is an interface that defines the mechanisms for accessing data in a document. This allows programmers to create dynamic content with standardized tags. Currently the DOMs used by MSIE and Netscape are proprietary, but both will be standardized with the next releases.
DTD	***Document Type Definition***. An SGML document type, e.g., HTML is a DTD.
HTML	***HyperText Markup Language***. Ok, smarty, you know what HTML is, but it is important to note where it is in the scheme of things. HTML is an SGML DTD with pre-defined tags. XML allows programmers to create other DTDs which can be parsed by a browser.
OFX	***Open Financial Exchange,*** created by CheckFree, Intuit and Microsoft in early 1997, is a unified specification for the electronic exchange of financial data between financial institutions, business and consumers via the Internet.
RDF	***Resource Description Format*** is a data-modeling language using XML syntax. RDF is a way of describing and accessing data, especially valuable for sites which include rapidly changing data. *Aurora* is Netscape's RDF implementation.
SGML	**Standard Generalized Markup Language** is the grandaddy of all structured document languages. XML and HTML are both based on SGML.
SMIL	**Synchronized Multimedia Integration Language** is an XML application which allows programmers to synchronize multimedia elements as they are served to browsers.
URI	***Universal Resource Identifier***, the XML equivalent of the Universal Resource Locator (URL).
WWDX	***Web Distributed Data eXchange***. Allaire, creators of popular development application ColdFusion, developed WDDX as a mechanism for exchanging complex data structures between application environments such as JavaScript, ColdFusion, Perl, ASP/COM and Java.
XLinks & XPointers	Also referred to as XML Linking Language (XLL), *XPointers* work hand-in-hand with *XLinks*. XPointer points to data, while XLink describes the relationship of that data with other data.
XML	***eXtensible Markup Language***
XML Schema	***Also XML-Data,*** Microsoft's suggested replacement for DTD.
XSL	***XML Style*** (or ***Stylesheet***) ***Language***

Online XML Resources

- XML.com (www.xml.com)
- 20 Questions on XML (builder.cnet.com/Authoring/Xml20)

- Webmonkey (www.hotwired.com/webmonkey/xml/)
- Cafe con Leche (metalab.unc.edu/xml/)
- Project Cool (www.projectcool.com/developer/xmlz)
- Microsoft (msdn.microsoft.com/xml/c-frame.htm#/xml/default.asp)
- Netsape XML Resources (http://developer.netscape.com/tech/metadata/metadata.html)
- Allaire WDDX FAQ, Resources and Websites (www.allaire.com/handlers/index.cfm?ID=5624&Method=full)
- OFX Resources (www.ofx.net/ofx/default.asp)

Special thanks to Karl Wehden of Electonic Ink in Old City for his invaluable help in sorting out the XML world.

Ric Kolenda *is the Founding President of the Philadelphia Area New Media Association (www.panma.org), a division of the Eastern Technology Council. He can be reached at* nm_news@kolenda.com*.*

Published in PhillyTech Magazine, *September 1999.*

Articles Home