Standards Today

A Journal of News, Ideas and Analysis

A publication of CONSORTIUM INFO.ORG
GesmerUpdegrove LLP

October–November 2009 Vol. VIII, No. 6

FEATURE ARTICLE:

XML and its Many Children: Bringing Order to a Digital World

Andrew Updegrove [1]

Abstract: *Prior to the advent of computers, information was necessarily stored in tangible media that was searchable and understandable only through visual examination. With the advent of the Internet, both the opportunity and the challenge of automated access to knowledge were magnified a billion times. In the 1990s, it became clear that the riches of digitized data could only be mined if elements of text could be identified in such a way that they could be readily exchanged between computer systems of any type without losing knowledge of their own format and structure. Moreover, by permanently "tagging" elements of text with semantic, as well as formatting information, the data in documents could become self-aware, allowing information to be more intelligently searched, manipulated, and compiled. The mechanism invented to achieve this end was a standard called the Extensible Markup Language (XML), a tool that was strict enough to achieve the interoperable exchange of information, but flexible enough to allow the creation of a derivative based language to order and make greater sense of any domain of knowledge. In this article, I describe the origins, development and impact of XML, and the standards development organizations that maintain and continue to develop these essential tools of the Digital Age.*

Introduction: Prior to the Internet era, the science of organizing, archiving and accessing information evolved at a leisurely pace. With all information fixed in tangible media, only the highest level of indexing made sense, lest the volume of index materials become too cumbersome to be useful. In due course, libraries adopted common conventions for indexing (via subject, author and title) and archiving (using the Dewey Decimal System) that provided useful, albeit slow and limited, ways for researchers to locate what they needed.

[1] Disclosure: the author and his law firm have acted as legal counsel to a number of entities mentioned in this article, including the Association for Cooperative Operations, Research and Development (ACORD), Open GeoSpatial consortium (OGC), Organization for the Advancement of Structured Information Standards (OASIS), and XBRL International, Inc. (XII).

Once computers became powerful and sophisticated enough to analyze the contents of entire books in nanoseconds, the need for research tools capable of capitalizing on this raw power became obvious. More was needed than simply old wine in new bottles, though, because the reach of the new technologies was orders of magnitude greater than the conceptual grasp of the card catalog metaphor.

The advent of the Internet, and more particularly the hyperlinking capability of the World Wide Web and the development of highly effective browser technology, raised the ante even more substantially. Soon, the volume of data becoming accessible presented a veritable Black Hole of knowledge in reverse, exposing gigabytes of new information to virtual access on a daily basis.

Simple access to data, however, does not equate to an ability to make practical use of that information. A way was needed to make the text of digitized documents transportable and able to be displayed with its original formatting intact on the proprietary computer equipment sold by every vendor. And in order for the information contained in electronic documents to be automatically extracted and combined with other data of a similar type, information of particular types needed to be identifiable as such (e.g., as annual profits, or lab test results or census data). Otherwise, we would simply drown in our own newly accessible data, no better off than before.

Making data practically useful therefore required the development of new technical means to make information machine-readable, so that computers could manipulate it most efficiently, in the first instance, and do useful things with it in the next. Since the goal was to be able to access, share and redisplay information anywhere on the Internet, data needed to remain human-readable as well. In short, making the data and knowledge of the world electronically accessible required new standards - and lots of them.

Simple access to data does not equate to an ability to make practical use of that information

Today, there are a variety of standards that deal with information on the Internet. But in many ways, the DNA that lies at the heart of categorizing, storing, accessing, manipulating and presenting electronic information is a single standard. Or, more properly stated, a single matriarch standard that has given birth to, and continues to nurture, an almost limitless number of specification descendants. That standard is the Extensible Markup Language – more familiarly known simply as "XML."

What is XML? Simply stated, it is a text format standard, with supporting tools, that allows structure and meaning to be given to electronic documents (broadly construed) through the use of machine-readable, standardized tags. Most importantly, XML is a flexible standard that not only allows a programmer to give digital meaning to words and sections of text, but also to decide what that meaning should be. The result is that everyone from biochemists to Old Testament scholars to ad writers uses XML to create languages unique to their needs.

In this article, I will review the origins and development of XML, the organizations that develop and maintain XML-based standards, and the explosive growth and influence of XML in the decade following its formal adoption in 1998.

I XML Origins and Development

XML did not spring full blown from the minds of the World Wide Web Consortium (W3C) Working Group members that created it. Rather, it evolved from a number of predecessor specifications that systemized ways of electronically annotating text to give additional, computer-readable meaning to text of various types (i.e., "this is a table," "this is a chapter heading," and so on). In effect, these precursors played a role similar to an editor's traditional blue-pencil marginal comments on pre-print text – and hence the name "markup languages." But unlike an editor's penciled marks on fixed copy, markup language text can remain(to human eyes) invisibly embedded forever, available to instruct future processors how to more usefully work with the text in question.

GML and SGML: The immediate predecessor to XML was the Standard Generalized Markup Language, or SGML, adopted by the international Organization for Standardization (ISO) in 1986. SGML, in turn, was based in part on an earlier specification called the General Markup Language (GML). That language was developed within IBM, beginning in the 1960s, by a team led by Charles Goldfarb, Edward Mosher, and Raymond Lorie.[2] The goal of the project was to create a robust means for governments and document-dependent private industries (such as the legal sector) to create large documents that would be able to retain their structure over long periods of time. But while each markup language relied on the innovations of its predecessor, each in turn also broke new ground.

In the first evolutionary step, GML's creators sought to provide structure, but not directions, to documents. For example, GML tags would identify appropriate text as a table or a section heading, but would not go on to tell a computer how a table or a section heading should be formatted. That was left to the specific software application that might be used to open and print the document. As a result, text could more easily be exchanged among users with different software, and those users could also make independent decisions on matters such as font choice, table size, and so on.

Goldfarb went on to become a member of a committee formed in 1978 and administered by the American National Standards Institute (ANSI), called the Computer Languages for the Processing of Text committee. Later, he was asked to chair the working group chartered to create SGML. The first working draft of SBML emerged in 1980, and in 1983 the sixth draft was formally adopted by the Graphic Communications Association as GCA 101-1983. Soon, government agencies like the U.S. Internal Revenue Service and Department of Defense were using the new standard.

Development work on SGML continued in the ANSI committee as well as within a new working group (SC18/WG8) organized under Joint Technical Committee 1 of

[2] It was no coincidence that the last name initials of the three engineers also happen to be "GML."

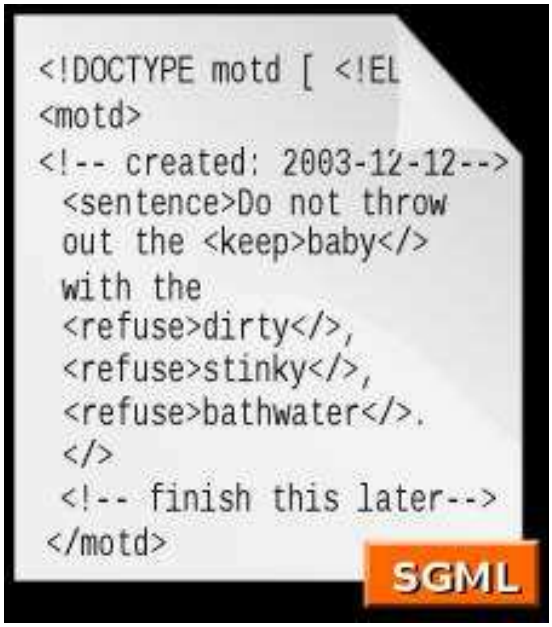
ISO and the International Electrotechnical Commission (ISO/IEC JTC 1). Goldfarb was project editor for each, and in 1986, the international SGML standard was published as ISO 8879:1986.

But while SGML was robust, it was also complex and restrictive. In order to conform to SGML, each document was required to conform to a "Document Type Definition" (DTD), made up of the "markup declarations" that the document could (and must) utilize. The result worked well for setting up templates for standardized documents, but the requirement to either use, or create, a DTD for each document was both limiting and burdensome.

While the advantages of SGML played well to a world of centralized computing systems managed by experts, the highly regimented standard was not nimble enough to meet the demands of the often self-trained, hacker-mentality developers that were building out the Internet and the Web. As noted by one on-line commentator not long after XML was formally adopted by the W3C:

> The Web culture is about "freedom". To most Web geeks, DTDs are just dull and boring stuff, intolerable obstacles to their creative freedom. Joe Webmaster wants bouncing logos, blinking commercials, 3D buttons, flashy fonts, background music like in Starwars. Joe Webmaster had extensive training with MS InterDev, JavaScript, ActiveX. He had no training with SGML and does not plan to…Joe Webmaster's freedom is immense. Do you want an indication of how immense it is? Just have a look to the shelves at your favorite computer store. *Awsome [sic]*, isn't it? (Then try to find the SGML books -- if there are any....) [3]

Moreover, some features that added flexibility to SGML became disadvantages on the Internet. For example, SGML could function as an umbrella standard for other markup languages, which could be referenced in the DTD for the document being created. The result was that not every processor would be able to read every SGML compliant document – a significant disadvantage when seeking to read a document available at a hyperlinked site.



In order to preserve the benefits of SGML while addressing these issues, a less restrictive subset language was needed that would allow, but not require, the use of DTDs, and address other matters of concern. But while some recognized the

[3] *See,* Sabarthez, Laurent, Some Notes on the History of XML (August 1998), *at*: http://www.users.cloud9.net/~bradmcc/xmlstuff.html All Web site cited in this article were last accessed on December 9, 2009.

need for such a standard, others did not, or believed that formalizing an SGML equivalent for Web pages was more important.

The eventual result was the development of two new standards – one inspired but not so faithfully derived from SGML, to be used for uniformly rendering Web pages (HyperText Markup Language, or HTML), and some time later, a second one, more narrowly based on SGML, for use with documents accessible via the Internet (XML).[4]

The development of XML: SGML traced its origins to the world of mainframe computers, and as a result it was developed within the leisurely, and often more exacting, process of the standard setting infrastructure that had evolved over more than a hundred years. The birth of XML, however, was far different.

Between the adoption of SGML and the chartering of the XML working group a revolution had occurred not only within the information technology (IT) industry, but within the standard setting community as well. Beginning in the late 1980s, IT vendors increasingly opted to form variously formal and informal new organizations to develop the new specifications they needed. Their motivations included a mix of frustration with the slow speed of traditional standard setting, as well as the desire to exercise greater control over the specifications that emerged from their combined efforts.

By the time the Internet began to be massively used, there were already hundreds of these "consortia" in existence, many of which had become institutionalized, as well as the centers of domain effort within their self-appointed areas of competence. As a result, while a great deal of IT standard setting continued within traditional standard setting organizations, the majority of activity, and hence the center of influence, in IT standard setting had passed to the consortium world.[5]

One of the most successful and respected of these new consortia was the World Wide Web Consortium (W3C), conceived and founded in 1994 by Web inventor (now Sir) Tim Berners-Lee at the Massachusetts Institute of Technology (MIT) Laboratory for Computer Science. The immediate inspiration for the effort was the evident need for a single, universally implemented HTML standard, but over time the W3C became the venue of choice for the development of a variety of standards directed at improving the effectiveness of the Web, and ensuring that its benefits could be shared throughout the world.

As a result, when the need became clear (to some) for a new version of SGML that would be optimized for use on the Internet, the W3C seemed like the logical venue in which the effort should be launched. But as the Internet became more economically important, the design of the technology underlying it became more strategic, and particularly so after the meteoric rise of Netscape Communications

[4] HTML, like XML, is a formatting language that allows information (both fixed text, as well as dynamic elements, such as video) to be displayed. Proper use of HTML in the creation of a Web page allows any browser that makes proper use of the same standard to display information as originally intended. The faithful use of additional standards ensures that those with vision, hearing or other disabilities will also be able to access the same information.

[5] The most extensive available list of active and inactive IT consortia is maintained by the author, and can be found at: http://www.consortiuminfo.org/links/ Over time, ISO/IEC developed processes that allowed consortium-developed standards to be submitted to, and approved by, JTC1 working groups, thus allowing these standards to gain the imprimatur of the traditional standards regime.

Corporation, which enjoyed one of the most successful initial public offerings in history on August 9, 1995. Suddenly, the question of what "SGML on the Web" should mean became a subject of significance, although only a few fully appreciated that fact.

Microsoft was one of those that did, in part because for some time it had famously missed the importance that the Internet and Web would assume. As a result, when Netscape came to own the suddenly hot market for Web browsers, Microsoft was left scrambling. In response, it mounted an urgent effort to counter the instant success of Netscape's Navigator browser with its own hastily launched Internet Explorer software. But while Microsoft came late to the Internet party, it recognized the importance that an Internet-optimized subset of SGML could play.

The result was the formation within the W3C of the "Generic SGML Editorial Review Board" in 1996, chaired by Jon Bosak, of Sun Microsystems. At the same time, a "Generic SGML Working Group," was formed, which in turn was supported by a Special Interest Group. The design goals of the new working teams were as follows: XML shall be straightforwardly usable over the Internet.

Incredibly, the first working draft of the XML standard was released only twenty weeks after work began

1. XML shall be straightforwardly usable over the Internet.
2. XML shall support a wide variety of applications.
3. XML shall be compatible with SGML.
4. It shall be easy to write programs which process XML documents.
5. The number of optional features in XML is to be kept to the absolute minimum, ideally zero.
6. XML documents should be human-legible and reasonably clear.
7. The XML design should be prepared quickly.
8. The design of XML shall be formal and concise.
9. XML documents shall be easy to create.
10. Terseness in XML markup is of minimal importance.[6]

As with all other W3C Recommendations, the final product would be distributed for free.

Work began in July of 1996, with James Clark as technical W3C Technical Lead, and Tim Bray and C. Michael Sperberg-McQueen as co-editors. Although the SGML standard did not provide the sole reference point for the new standard (the Working Group also borrowed ideas from the HTML and HTTP standards, among other sources), XML was primarily intended to be a selective, narrower profile of SGML.

Incredibly, the first working draft of the XML standard was released only twenty weeks later – on November 14, 1996. The final version of XML 1.0 was formally adopted as a W3C "Recommendation" (i.e., standard) on February 10, 1998. Along the way, there were intense efforts by both individuals (with strong feelings on technical matters), as well as competing companies (with huge investments riding on their ability to successfully navigate the rising tide of the Internet), to influence

[6] "1.1 Origin and Goals," Extensible Markup Language (XML) 1.0 (Fifth Edition) *at:* http://www.w3.org/TR/REC-xml/#sec-origin-goals

the final form of the standard.[7] In all, eleven individuals comprised the original working group, laboring through weekly teleconferences and via email, while as many as 150 others participated in the active email discussions via the working group listserv.

Following the adoption of XML 1.0 the W3C (and the wide adoption of the standard in the field), XML development work was expanded and restructured under the direction of a newly chartered XML Coordination Group and XML Plenary Interest Group. The actual design work would now be conducted in five new XML working groups addressing topics such as XML Schema and XML Syntax; internal liaison relationships were established with other W3C working groups active in technically adjacent areas to ensure overall architectural coherence.

Despite its rapid development, XML proved to be remarkably durable. Today, XML 1.0 is in its fifth edition, but the changes to it have been minor. A version 1.1 of XML was adopted in February of 2004, but its use has been less widespread. The stability of the standard is due in part to the fact that XML has proven to be robust and useful, and also because, once implemented, any standard is difficult to change without introducing incompatibility problems with documents and applications already created.

What it is: In concept, XML is disarmingly simple. Like its partner in digital presentation, HTML, XML employs "tags" to separate content and assist computers in more usefully and easily dealing with that data. Tags are identifying labels contained within angle brackets, as follows: "<tag>." There are three basic types, two of which are used to enclose the content to which they relate (called start tags and end tags, such as <address> and </address>) and "empty element" tags (e.g., <line-break/>), which separate content. Tagged content, and empty element tags, can be nested within and between other tags, permitting the marking up of content within content (e.g., minor heading content within major heading content).

Tags can also serve other purposes, such as assigning "attributes" to text by pairing tag information of one type with a value. For example, tags can not only identify a line of text as a section of a document, but also assign a section number to the text enclosed by the start and end tags. Unlike the section number that is visible to the reader, but meaningless to a computer program, the value associated with a section heading tag can be machine-readable. This allows the software application in which the document is composed to take appropriate actions in relation to the tagged information, such as automatically adding a similarly numbered section line to the index of the same document

Tags can also add important search capabilities to documents, by giving machine-readable meaning to individual words. For example, in an XML language for legal documents, a party can be designated as the plaintiff rather than the defendant. A

[7] For a highly personal account of the rough and tumble development of XML and profiles of the individuals involved, *see* XML is Ten Years Old Today, posted by XML co-editor Tim Bray to his blog in 2008 at http://www.tbray.org/ongoing/When/200x/2008/02/10/XML-People. Some of the sharpest elbows were thrown when Bray took a consulting position with Microsoft arch-competitor Netscape midway through the development process. According to Bray, Microsoft insisted that he be removed as a co-editor; in an eventual compromise, he retained his role, but Jean Paoli, a Microsoft employee, was added as a third co-editor.

legal search using this "metadata" can therefore find (for example), only cases where Company A is a plaintiff, and not a defendant, and where Company B was a defendant. Tags can also indicate the technical nature of content (e.g., a picture would be identified with the tag "<img>." A search of a news archive could therefore be performed to find only pictures of a specific person in a boat, and exclude all text references to the same individual, whether or not in a boat. The tagged data in question therefore "knows" what it is, and can identify it as such to a search function.[8]

Tags take advantage of another fundamental standard called Unicode. The Unicode is the product of a long-running project with the mission of ultimately making every character set of the past, present and future machine-readable.[9] XML tags therefore are allowed, with limited exceptions, to include only Unicode characters.

When a programmer properly adds XML tags to an otherwise appropriate document, the result is a file that can properly be read by any software application running on any operating system that has itself been developed in a compliant manner. But while the result is easily read by a computer, it is not so obviously interpreted by anyone not skilled in the programming arts. The following is a very simple example taken from the Wikipedia:

When a programmer adds XML tags to a document, the result is a file that can properly be read by any compliant software application running on any operating system

```
<?xml version="1.0" encoding='UTF-8'?>

<painting>

  <img src="madonna.jpg" alt='Foligno Madonna, by Raphael'/>

  <caption>This is Raphael's "Foligno" Madonna, painted in

  <date>1511</date>-<date>1512</date>.</caption>

</painting>
```

When interpreted by a work processing application, the above code would display an image of a painting by Raphael, together with the caption "This is Raphael's 'Foligno' Madonna, painted in 1511-1512." Moreover, the document would "know" (and therefore a search function could be informed) that the image was of a painting, that "Foligno" in this case (because the caption lies within the start and end tags for "painting" was the name of the painting, and that the work of art was created during the years noted – something not otherwise possible without the added information provided by the tags.

[8] A far more ambitious effort to improve the abilities of computers and search engines to automatically perform searches more intelligently is the long-ongoing Semantic Web effort of the W3C.

[9] The staff of the Unicode represents a group of unsung heroes doing yeoman service for the betterment of all mankind. For more on the significance of the Unicode, see my blog entry, The Unicode Standard 5.0: an Appreciation, *at:* http://www.consortiuminfo.org/standardsblog/article.php?story=20061017163856508

Unfortunately, many tags are not so intuitively named. As a result, properly tagged text, and especially highly formatted text, rapidly becomes incomprehensible to anyone other than a programmer. The following is a sample of a simple document header instruction:

<header [level="1-6"] [class="test"]>Heading/title</header>[10]

***Complexity and purpose**:* XML formats can be as complex or as simple as the task at hand requires. They can also be used for a variety of different purposes. Both points are well illustrated by one of the major standards wars of the last decade, which played out across two consortia, ISOIEC JTC1, and the national standards committees of scores of countries around the world.

The subject material for the drama was the ubiquitous office productivity software suite, which includes word processor, spreadsheet, slide presentation, and database modules, each of which is expected to be able to exchange data with the others. With such heavy formatting needs, the effort required to create a robust standard capable of preserving so much detail can be great indeed. Nevertheless, over time, there was increasing consensus that XML-based formats should provide the foundation for each of these modules.

The value of such a conversion was clear, for both internal as well as external reasons. For a proprietary vendor like Microsoft, the conversion of its Office software suite to an XML-based format would make it easier for Microsoft to upgrade its products in the future, and also make it simpler for the many independent software vendors (ISVs) that are part of the Microsoft Office ecosystem to keep their own products interoperating successfully with new versions of Office. This would in turn make it easier for customers of Microsoft and Microsoft ISVs alike to exchange information among the products they purchased from each.

As a result, Microsoft began to move away from its historic, binary formats to a new XML-based format that it named Office Open XML format, or OOXML. Microsoft offered OOXML to ECMA, a European-based consortium in 2006. After adopting OOXML, ECMA proposed the resulting specification to ISO/IEC for adoption the following year. In 2008, OOXML became ISO 29500.

But a second XML-based document suite format standard, called OpenDocument Format (ODF), had already been created by a consortium called OASIS (discussed in greater detail below). ODF was approved by the members of OASIS in 2005, submitted to ISO/IEC JTC1 later the same year, and adopted as ISO 26300 in 2006. The ODF standard had been created for quite a different purpose: to allow multiple, independent office suites (and other types of software) to coexist, each

---

[10] While simpler than SGML, XML is still no treat for newbies. *See*, for example, this sentiment from a book on coding using XML and Perl (another programming language):

> Many people, understandably, think of XML as the invention of an evil genius bent on destroying humanity. The embedded markup, with its angle brackets and slashes, is not exactly a treat for the eyes. Add to that the business about nested elements, node types, and DTDs, and you might cower in the corner and whimper for nice, tab-delineated files and a split function.

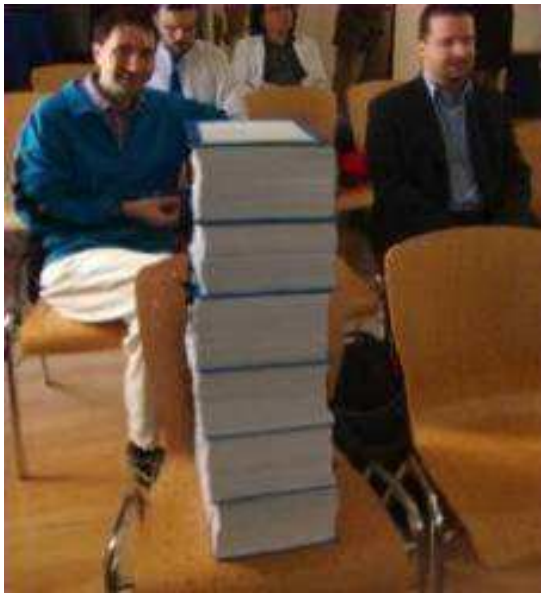
Ray, Erik T. and McIntosh, Jason, Perl and XML, Section 1.2 (O'Reilly 2002) *at*: http://docstore.mik.ua/orelly/xml/pxml/ch02_01.htm.

able to exchange documents with the other. Customers could thus choose among a variety of competing, but compliant, products without concern over being "locked in" to the products of any vendor, since they could easily move to the offerings of a competing vendor any time they wished.

Each standard was therefore intended for the same general purpose – to allow information to be exchanged within documents and software products, and for documents to be exchanged between systems, in each case without loss of data integrity or formatting. But since the business goals underlying the creation of each standard were different, the resulting standards were as well.

Because ODF was intended to enable documents to be exchanged between a wide variety of software products, both desktop based as well as remote (i.e., "in the cloud"), and proprietary as well as open source, the creators of ODF sought to strike a traditional balance between level of detail and freedom to innovate above the level of standardization. As a result, the final specification was c. 700 pages long – quite lengthy for a standard of any type, but most standards are not required to address such a long and detailed list of parameters.

OOXML, on the other hand, had a different goal: to permit the faithful replication of every aspect of a single proprietary product - Microsoft Office - down to the finest detail. The result was a specification that weighed in at over 6,000 pages, filling six binders that, piled one atop the other, stood four feet high.



The OOXML Standard Stack

Happily for all, most XML languages and formats can be far shorter in length. But the example of ODF and OOXML nonetheless illustrates the extremely wide range of requirements to which XML can accommodate.

Schemas and more: While XML was intended to be a narrower and more restrictive version of SGML, over time it replaced SGML for almost all purposes, both on and off the Internet. In a recursive twist, the` W3C created its own narrower version of the SGML DTDs that had helped inspire the XML development effort to begin with. The specification for creating what were called "XML Schemas" became a W3C Recommendation in May of 2001, and enabled the creation of shared XML vocabularies and rules to define the structure, content and semantics of an XML document. True to the original spirit of creating XML as a simpler subset of SGML, far more XML documents continued to be created without reference to schemas.

Today the XML environment is supported by a variety of other W3C efforts, including 10 standing working groups.[11] These working groups create standards

[11] Current W3C XML Working Groups are listed at the home XML page, found here: http://www.w3.org/XML/ The status of current efforts are listed at the XML Activity Page, found here: http://www.w3.org/XML/Activity

and tools intended to make XML documents more useful, such as tools to assist in formatting, exchanging and searching XML documents.

II The World of XML Implementation

The advent of the Digital Age has presented information-handling challenges at every level. But the inescapable need to wrestle with issues such as formatting and the reality of proprietary hardware and software has also offered a unique opportunity to increase the ability to exchange information across all languages, cultures and distances. Through the steady progress of the Unicode, for example, a single computer can work with documents composed not only in modern German, Japanese and Arabic, but in ancient Sumerian and Babylonian as well – each in their own unique scripts.

XML likewise provides the means to take a major step towards putting the vast archive of human history and creativity into a more universally usable form. It does so by providing, not a single language, but the computer-linguistic (as it were) tools to create an infinite number of domain-specific languages. The response in the marketplace has been nothing less than phenomenal.

The secret behind the success of XML lies in its simultaneous rigidity and flexibility. Rigid, because the way that XML tags are created, used and read must remain the same so that software can use a single methodology to address them. But flexible, because anyone can create a new XML language, for whatever purpose she may wish, and generic computer systems will nonetheless be able to work intelligently with those tags.

A language for every purpose: The result has been an explosion of efforts to create custom tag sets comprising new XML languages that can be used to work more efficiently with data of any kind imaginable, from sports information to periodical advertising to financial information. Most significantly, tags can be used not only to designate factual data, such as street addresses and section headings, but anything else as well. Once labeled, information can be selected, manipulated, combined, and displayed more intelligently.

The use of XML permits the automatic population of charts, tables and spreadsheets with data that otherwise would need to be extracted, totaled and manually reentered as a separate step. Moreover, using XML related tools (such as XML Query), XML tagged data can be combined not only from text documents of a similar type, but from all of the following at the same time: text documents, databases, Web pages, and spreadsheets.

XML can be used to address the narrowest of niches as well as the broadest. At the universal end of the spectrum, one can find XML languages such as XBRL, created to permit multinational and national enterprises the world over to identify and present financial data in a uniform fashion, thus enabling global financial information to be more easily consolidated within multinational corporations, and for regulatory filings to be made and examined more easily. The narrow end of the spectrum is suggested by the following excerpt from an article that appeared in the financial press less than three years after the formal approval of XML:

> XML has taken a strong foothold in the financial services industry, and the weather derivatives market is next in line for a standard trading protocol of its own. The Weather Risk Advisory, a software and consulting company focusing on weather derivatives, is leading an initiative to develop WeatherML, an XML-based data protocol that looks to be a standard for the electronic processing of weather derivatives….
>
> Currently,…weather derivatives trading [is] a tedious and manual process for all parties involved. "For example, trader A has to enter the details of a transaction on the system and the system holds an internal representation of that transaction….Counter parties to that trader would have their own systems with internal representation of the transactions data." In other words, there is no standard to integrate the two systems and foster automatic communication between trading parties. This manual system, in turn creates operational risks with the re-keying of information into each separate system by the traders. "Once in place, WeatherML would allow straight-through processing and the users could connect components within their overall architecture to each other in a seamless manner,"…[12]

XML development organizations: While XML languages can be created without enormous effort in most cases, they only become broadly useful if they are widely implemented. Since pervasively implementing an XML language represents a substantial commitment by a document creator, a new language therefore needs to have credible support from the outset in order to be successful. In order for that to happen, potential implementers will look to whether the language will be maintained (and ideally promoted) over the long term by a credible organization. The result has been the development of a multi-tiered ecosystem of standards consortia. The following is a selection of currently active consortia, grouped by the role that they play in supporting both the XML ecosystem and those that are dependent upon it.

Foundational XML organizations: At the top of the stack are two principle organizations: the World Wide Web Consortium, which developed and maintains the core XML standard and related broadly applicable standards and supporting materials, and OASIS, the consortium that has been most active in developing a broad spectrum of standards, guidelines, profiles and tools based upon XML to facilitate eCommerce across multiple business domains.

- ✓ ***World Wide Web Consortium (W3C)****:* As already noted, the W3C was formed to ensure that the Web evolved in a more orderly and standardized fashion. While its initial efforts focused on HTML, it's work program

[12] McEachern, Christina, A New XML-based Standard for Weather Derivatives Transactions Proposed, Wall Street Technology, December 22, 2000, at: http://www.wallstreetandtech.com/technology-risk-management/showArticle.jhtml;jsessionid=XFR3LRZCN1XU3QE1GHRSKH4ATMY32JVN?articleID=14704626&_requestid=629891

expanded into a variety of areas supporting the Web, including XML, Web Design and Applications, Web Architecture, Web Services, Web Devices, and Browsers and Authoring Tools. It has also dedicated significant resources (and a great deal of missionary effort by Tim Berners-Lee) to develop and promote a new set of standards directed at enabling a new and richer layer of meaning to Web-hosted information.

The goal of that effort is to make possible the evolution of a "Semantic Web" of information, using a more sophisticated set of tags and tools to invest Web hosted information with more pervasive machine-readable information. As with XML, a Semantic Web document would identify data with additional attributes of various types, such as geographic location, business type (e.g., a theatre), and more (such as the theatre's hours of operation), so that users of the Web could perform far more sophisticated and useful searches (e.g., "find any Chuck E. Cheese restaurant in city X within three blocks of a theatre showing Willy Wonka and the Chocolate Factory between 1 and 5 this afternoon").

As of this writing, W3C a total of 143 standards, guidelines and other deliverables (including serial versions of the same material) have been adopted by W3C members as formal Recommendations, with many more under development.[13]

- ✓ ***Organization for the Advancement of Structured Information Standards (OASIS):*** Formed in 1993 as SGML Open, OASIS was initially created to promote SGML rather than to undertake technical activities of its own. With the advent of XML, OASIS changed its name, and also began to undertake XML-based standards development efforts of its own. Unlike consortia that remain focused on a single standard and a limited work program, however, OASIS adopted a more open, "Big Tent" technical process philosophy that allows a small number of members to launch a working group effort within the broad perimeter of the overall OASIS mission. As a result, by 2004, there were 70 working groups in operation, and the number of activities in process at any time has remained high.

  Broadly stated, as W3C is to foundational XML tools, so OASIS is to developing XML languages and related tools to address specific domain needs. Its reputation in this area was augmented soon after the launch of XML when UN/CEFACT, a United Nations committee concerned with business standards, selected OASIS as its partner to develop XML-based standards to serve the evolving needs of eCommerce. The result was the eXtensible Business Language, or eBXML, which later became ISO 15000.

  Since then, OASIS has developed a wide variety of XML-related specifications serving the needs of commerce over the Internet, both broad and narrow. Examples of the broad variety include:

  - o Universal Business Language (UBL): UBL comprises a library of useful, standard electronic forms such as purchase orders and invoices that

[13] The main information page for the W3C can be found at: http://www.w3.org/Consortium/ The main standards page can be found at: http://www.w3.org/standards/

can be easily integrated with existing software without additional data entry

- Security Assertion Markup Language (SAML): SAML allows the practical exchange of authentication and authorization information for the benefit of a Web site user and a Web host through the use of a third party identity services provider, thereby enabling not only more secure use of the Internet, but also a "single sign on" convenience for the user.

More recently, OASIS has developed XML-based standards for a variety of other purposes and constituencies, such as the Common Alerting Protocol, to facilitate the rapid dissemination of emergency warnings, standards to enable the Smart Grid, and an end to end suite of standards and processes to facilitate electronic voting, including the Election Markup Language.

As of this writing, OASIS supports 83 adopted standards (again, including multiple versions of the same standard).[14]

Single Focus organizations: Many consortia have been formed to develop XML languages and related tools for use in a single industry sector, or to meet a specific need of businesses generally. The following is a very small, but representative sampling:

- ✓ ***International Press Telecommunications Council (IPTC):*** While the IPTC is heavily invested in the development and support of XML based standards, its standards development activities predate the existence of XML by two decades. Given its mission of developing standards for the interchange of news data, it is not surprising that XML-related efforts now represent the backbone of its efforts. Today, the IPTC supports a suite of XML-based languages, each tailored to the needs of working with a specific type of news data. Those standards include:

  - NewsML-G2: An XML-based general purpose exchange standard able to deal with news of any kind and media of any type

  - EventsML-G2: An XML-based standard for exchanging event-related data in a manner conducive to news reporting

  - SportsML: An XML-based format tailored to sports statistics and other information

- ✓ ***HR-XML Consortium:*** HR-XML, as its name suggests, was created to develop and promote standards for use by human resource professionals across all industries, based on XML. More specifically, its mission is to

[14] Adopted OASIS standards can be found here: http://www.oasis-open.org/specs/ OASIS also sponsors the Cover Pages Web site, the most exhaustive resource on the Internet relating to all things XML. The Cover Pages site can be found at: http://xml.coverpages.org/

develop a "standard suite of XML specifications to enable e-business and the automation of human resources-related data exchanges."

- ✓ ***XBRL International, Inc. (XII):*** XII develops and maintains the eXtensible Business Reporting Language (XBRL) specification for global use in financial reporting, utilizing multiple W3C standards (i.e., XML Schema, XLink, XPath and Namespaces) in order to permit the degree of highly structured presentation of financial data that financial reporting requires. XBRL utilizes metadata included in taxonomies that define reporting concepts and interrelationships between concepts and semantic meaning. XBRLS, a simplified application profile of XBRL, allows non-XBRL experts to create XBRL metadata and reports. In order to permit both localization as well as uniform reporting tools for multinational corporations, XII is based upon a national membership model, with each "jurisdictional" member in turn having domestic members. Representatives of these members may in turn participate in XII Working Groups.[15]

XML-adjunct organizations: XML provides a useful mechanism that many consortia use in connection with some of their standards activities. While less obviously identifiable as "XML consortia," they nevertheless provide XML languages and other tools of importance to the marketplace, either in specialized areas, or more broadly, as befits their overall mission.

- ✓ ***Association for Cooperative Operations, Research and Development (ACORD):*** Unlike the other organizations discussed in this article, ACORD is an American National Standards Institute (ANSI) accredited standards development organization. While it does include technology companies as members, its core constituencies are insurance underwriters, brokers and other commercial enterprises in the insurance sector. As a result, rather than creating XML languages to be used by others to create products, it uses XML to create useful forms, frameworks, and guidelines that are directly usable by its members in their businesses, or which can be incorporated into their own internally generated tools.

- ✓ ***Internet Engineering Task Force (IETF):*** The IETF is one of the foundational consortia enabling the Internet, transitioning in 1991 from a government project to one with public participation. Famous for its "rough consensus and running code" philosophy, it has been well suited to developing and maintaining some of the core standards upon which the Internet is based, including the Transmission Control Protocol and the Internet Protocol (TCP/IP). When Web site syndication (i.e., the ability to be notified when new material is posted at a given Web page) gained in popularity, the IETF chartered a new XML-based activity to improve upon the RSS syndication feed, called the Atom Publishing Format and Protocol (AtomPub) Working Group.

[15] Links to XBRL International taxonomies, specifications and best practices documents can be found on the left side of the XII home page, at: http://www.xbrl.org/Home/

- ✓ **Open *GeoSpatial Consortium (OGC):*** For the last fifteen years, OGC has served as the primary venue for standards activity addressing the rapidly evolving needs of government, defense, agriculture, science and many other domains to work efficiently with geospatial information. Its 28 (to date) adopted standards include many that are based upon XML, including the following:[16]

    - o Geography Markup Language Encoding Standard (GML): A grammar (schema and instance document) for expressing geographic features

    - o City Geography Markup Language (GML): A schema of OGC's GML adapted for 3D city and landscape models

    - o Geospatial eXtensible Access Control Markup Language (GeoXACML): An extension to the OASIS eXtensible Access Control Language (XACML) allowing the incorporation of spatial data types and spatial authorization decision functions.

    - o Keyhole Markup Language (KML): Based upon a contribution from Google, this standard, is intended to standardize the use of geospatial data in on-line 2D maps and 3D earth browsers.

    - o Sensor Model Language Encoding Standard (SensorML): SensorML specifies models and XML encoding to permit the geometric, dynamic, and observational characteristics of sensors and sensor systems to be defined, from simple visual thermometers to complex electron microscopes and earth observing satellites.

Commercial languages: While XML languages are usually created by non-profits membership organizations, they can also be created and sold by commercial enterprises. A rather exotic example is the Spacecraft Markup Language, developed by SRA International, Inc., for the aerospace industry.

Other: The ease with which XML languages can be created and the appeal that such an exercise has to specialists with programming skills as well as professional software developers has resulted in an astonishing array of efforts, some of which have been transitory and others sustaining. The following is a very short excerpt from an impressively long list compiled, but not recently updated, by Robin Cover at the CoverPages Website. The examples below are meant to suggest the breadth, rather than (in some cases) the depth, of the efforts listed:

Chemical Markup Language
XML Common Biometric Format (XCBF)
Signed Document Markup Language (SDML)
Real Estate Transaction Markup Language (RETML)
Emergency Data Exchange Language (EDXL)

[16] The main standards page for OGC can be found here: http://www.opengeospatial.org/standards/tml

Mathematical Markup Language (MathML)
vCARD in XML and RDF (Electronic Business Card)
Historical Event Markup and Linking (HEML)
Telecommunications Markup Language (tML)
Robotic Markup Language (RoboML)
Physics Markup Language (PhysicsML)
Exploration and Mining Markup Language (XMML)
Navigation Markup Language (NVML)
Astronomical Markup Language
AdMarkup XML DTD for Classified Advertising
Printing Industry Markup Language (PrintML)
Tutorial Markup Language (TML)
SpeechML
Architecture Description Markup Language (ADML)
Theological Markup Language (ThML)
OpenText.org Papyrus Encoding Markup
LitML: A Liturgical Markup Language
FlowML: A Format for Virtual Orchestras
Staffing Industry Data Exchange Standards (SIDES)
Electronic Thesis and Dissertation Markup Language (ETD-ML)
Steel Markup Language (SML)
Marine Trading Markup Language (MTML)
Chess Markup Language (ChessML)
Mind Reading Markup Language (MRML)[17]

III The Future

As XML co-editor Tim Bray observed in the W3C press release celebrating the tenth anniversary of the adoption of XML, XML will not be the last platform-independent, vendor neutral standard that will be needed to manage the ever-expanding flood of data that we continue to create. New standards will be required to make better and more efficient use of data on the Web, and perhaps more sophisticated standards will be needed to manage the data itself as the volume and nature of that information changes.

But given the enormous amount of information that is already exposed to the Web, it will be far more difficult to implement new standards than it was with XML, when the Web was still young, and material was still being prepared for on line accessibility for the first time. To some extent, this challenge may be ameliorated by automatic tagging tools. But experience to date with the W3C's long campaign to inspire implementation of its Semantic Web standards demonstrates that the benefits of new tagging or other systems will need to be very demonstrable before broad implementation can be expected.

That being the case, the incredible success of XML becomes meaningful in a new way: as the standard we have, we need to commit to use it most effectively. The

[17] The complete (and seemingly endless) CoverPages list of XML Applications and Industry Initiatives can be found at the CoverPages Web site at: A more current, but much less entertaining, list of XML languages can be found at the Wikipedia, at:
http://en.wikipedia.org/wiki/List_of_XML_markup_languages

success of XML also makes obvious the enormous benefits that properly conceived and executed information standards can bring. Hopefully, this will provide the incentive to make the substantial investments that may be needed to retrofit the Internet and the Web with the new standards that will inevitably follow, and that will further enrich our experience of all that the Digital Age has to offer.

Sign up for a free subscription to ***Standards Today*** at

http://www.consortiuminfo.org/subscribe/2.php?addentry=1

Standards Today is a free bi-monthly electronic Journal sponsored by the Boston law firm of Gesmer Updegrove LLP. The current issue of ***Standards Today*** and a subscription form may be found at www.consortiuminfo.org/bulletins. Questions or comments about the articles in this issue or about ConsortiumInfo.org may be directed to Andrew Updegrove at updegrove@consortiuminfo.org, or by telephone at 617/350-7800. To learn more about Gesmer Updegrove's standards and open source practice, visit http://www.gesmer.com/practice_areas/consortium.php, or contact Andrew Updegrove.