# The XML Files: The Truth Will Be Out There

**This document was prepared by Cara Bradley to meet the requirements for LIS 538, Automation in Libraries, at the School of Library and Information Studies at the University of Alberta.**

The World Wide Web has become an integral part of library and information centers, providing access to a wider range and variety of information. While librarians and information professionals naturally focus their attention on the content derived from the World Wide Web and other sources, the technology underpinning this information is of vital importance. Librarians and information professionals must keep abreast of new developments that will affect the way they deliver information. Hypertext markup language (HTML) has been the standard means of delivering information over the World Wide Web since its widespread popularization earlier this decade. However, new markup languages are appearing that will change the way that information is delivered over the World Wide Web. Standard Generalized Markup Language (SGML) is already being used in a number of library and information contexts; however, its complexity and cost have limited its appeal. EXtensible Markup Language (XML), a simplified version of SGML, has begun to receive a great deal of attention on the World Wide Web. The potential of XML to revolutionize library and information centers is also receiving some attention. This paper is an attempt to introduce librarians and information professionals to eXtensible Markup Language. The paper has been divided into the following sections for easier navigation:

**Brief History of Markup Languages**

- HTML and SGML

**XML**

- Introduction
- Comparison of XML with HTML and SGML
- The XML Family- XML, XLL, and XSL
- Structure of XML
- Validity and "Well-formedness"
- HTML and XML
- Potential Applications of XML
- Benefits of XML
- Drawbacks of XML
- Current State of XML
- The Future of XML

**XML and Libraries**

- Introduction
- Databases
- Other Potential Applications
- XML Projects of Interest to Librarians

**Conclusion**

**Bibliography**

- XML in Library/Information Environments
- XML - General


---

# History of Markup Languages

## HTML AND SGML

Markup languages such as HTML, SGML, and XML use tags or other indicators to instruct a computer how to display, recognize, or otherwise manipulate marked material. Hypertext Markup Language (HTML) is the most widespread markup language, and most librarians have undoubtedly had some exposure to HTML, either through production of HTML documents, or simply through experience retrieving information via the World Wide Web. HTML's widespread use has undoubtedly arisen because of the language's simple structure which, while it affords the user few options, is very easy and therefore affordable to use. HTML is presentation-oriented; it prescribes the appearance of text on the screen. It is driven by a single, finite tag set, which limits its functionality to those functions determined by the set tags. HTML's finite set of tags, as well as its exclusive focus on the appearance of the text, severely limit its usefulness. Sperberg-McQueen describes HTML as "bearable but limited" for describing screen layout ("XML and What It Will Mean for Libraries"). Other limitations of HTML include its inability to allow markup of specialty data formats, browser wars resulting from different formats of HTML from different vendors, and the rigidity of its conventions which prohibits user-defined tags (Halbert). HTML's real limitation is its inability to describe the content of a document. Sperberg-McQueen describes HTML as built "for publication-not for information retrieval, and not for scholarly study" ("XML and What It Will Mean for Libraries").

In 1986, the International Standards Organization approved the standard for Standard Generalized Markup Language (SGML), which defined the rules for creating textual markup languages. Unlike HTML, which is preoccupied exclusively with the appearance of text on the screen, SGML accommodates document structure and content in its markup tags. SGML overcomes the rigidity of HTML tags by allowing a user-defined Document Type Definition (DTD) and user-defined tags. However, the increased functionality of SGML comes at a price; SGML is very complex and difficult to use, and its "parsers are large and hard to reproduce and therefore tend not to be included in other pieces of software like Web browsers" (Sperberg-McQueen,"XML and What It Will Mean for Libraries"). SGML is complex and expensive to use, and the SGML specification itself is over 300 pages long (Boeri and Hensel). These factors have prevented SGML from achieving the widespread popularity of HTML;

while there are some projects that use this standard, SGML has never come close to achieving the acceptance characteristic of HTML. Thus, the potential benefits of SGML, including more precise searching and retrieval of information, have remained largely untapped.

Top

---

# XML

## Introduction

Thus, the World Wide Web has been increasingly restricted by the limitations of HTML and the complexity of SGML. XML was developed in 1997 in an attempt to find a middle ground between the extremes of these other two markup languages. XML is an emerging standard for markup languages, an attempt to facilitate the use of SGML-based document types on the World Wide Web. Unlike SGML, XML was not developed by the International Standards Organization (ISO), but by a consortium of nearly 300 companies under the guidance of the World Wide Web Consortium (W3C) (Boeri and Hensel). The XML draft standard was first proposed in November 1996, and XML 1.0 was released in February 1998 as a World Wide Web Consortium recommendation (Exner and Turner). Since this time, XML has received a lot of attention and is viewed by some as the tool that will revolutionize the WWW.

## Comparison of XML with HTML and SGML

XML, like HTML, originates from the Standard Generalized Markup Language (SGML). However, unlike HTML, XML is "eXtensible" in that it can be extended to suit many purposes. In fact, despite its name, XML isn't really a language. Instead, it is a metalanguage, or "a set of rules governing the development of unique tags for encoding documents" (Exner and Turner); its elements are not pre-defined but instead are defined by the user. XML can be modified to meet the demands of the user; while with HTML, the user worked within the parameters of pre-defined tags, the XML user can modify the markup language as needed. Unlike HTML, XML allows for adjustment and growth as the user's demands and the evolving WWW necessitate. XML also differs from SGML; while SGML is complex, and requires expensive software, XML preserves the extensibility and adaptability of SGML with less complexity and cost. The close relationship between the two standards is evident in the fact that all valid XML documents are also valid SGML. Sperberg-McQueen calls XML "SGML in disguise" ("XML and What It Will Mean for Libraries") and others refer to it as "SGML-lite" because the XML specification is less than a tenth the length of the SGML specification, and is thus much simpler to implement.

## The XML Family- XML, XLL, and XSL

At this point, it is probably wise to clarify the somewhat cloudy definition of XML. XML is both the name of a specific standard, and an umbrella term for the "XML family [which] includes several sections that address different functions of markup" (Exner and Turner). XML, eXtensible Markup Language, is the core of the new standard, governing "the semantic and syntactic structures of the data" (Exner and Turner). As Boeri and Hensel note, "XML by itself can be used to model and deliver structured data

without any reference to documents, and that may prove one of its earliest uses" ("XML"). However, for XML to have any widespread use, its style and linking standards must also be defined. Thus, the second standard covered under the umbrella term XML is eXtensible Linking Language (XLL). XLL is intended to enhance hypertext links. Rather than providing one-to-one paths between documents, XLL will take advantage of XML structure in order to facilitate links between various anchor points in documents. Boeri and Hensel provide an example that may clarify the role of XLL:

> Today, clicking on a link such as "Drug Family" could transfer you to a specific anchor point in an aspirin medical document; clicking on a link such as "Drug Interactions" might take you to a different anchor point in that same aspirin document. In XLL, clicking on the link could let you jump to a pop-up list of sections in the aspirin document, such as drug interactions, drug family, warnings, or other sections ("XML").

XLL will take advantage of the potential of hypertext by allowing new options such as multidirectional links. XSL, eXtensible Style Language, completes the triad often covered by the umbrella term XML. XSL governs the formatting and display of XML information. Elements like color, size, font, attribute, etc. are governed by XSL. Style sheets are advantageous because, unlike HTML, they separate style from content, allowing "regulation and consistency of style" (Exner and Turner). XSL will complement XML's ability to allow authors to specify their markups by allowing increased specialization of style for different types of documents (Exner and Turner).

## Structure of XML

The structure of XML is not unlike that of SGML and HTML. XML uses the angled brackets popularized by HTML. Start tags are content identifiers enclosed in angle brackets and end tags feature the backslash used for similar tags in HTML. Empty elements are indicated by a backslash after the content identifier. Name characters are all Unicode letters and, unlike HTML, are case-sensitive (Sperberg-McQueen, "XML and What It Will Mean for Libraries"). Empty elements are self-identifying; that is, they can be recognized without the DTD. Mixed-content problems are alleviated by content models, including ANY, EMPTY, and MIXED (Sperberg-McQueen, "XML and What It Will Mean for Libraries"). Thus, XML expands on many of the conventions popularized by HTML in order to ease the transition to the new standard while offering increased functionality.

XML's greatest difference from HTML, and the element which largely facilitates the increased functionality of XML, is the Document Type Definition (DTD). DTDs are a concept from SGML, but the XML DTD is not as complex as the SGML DTD. The DTD names the parts of a document and also specifies the order in which they are to appear. The DTD tells the processor what set of rules to use in deciphering the document. Because authors can define their own tags, the document needs "a declaration at the beginning to inform the receiving application how the tags are to be read" (Exner and Turner). A unique Document Type Definition can be established by the user, or the document can simply refer to an external DTD. Both types of DTDs contain the same kind of information, but the external DTD is a separate file to which the code in the document refers. It seems likely that most XML authors will refer to an external DTD, as composition of a Document Type Definition can be time-consuming and tedious. However, Exner and Turner note that an "organization using XML could make a company-wide DTD to be used in all its documents, so individual authors would not have to make their own" ("Examining XML"). This would allow the company to take full advantage of XML by specifying the form of XML

best suited to their needs, while also providing organizational consistency. Not all XML documents require a DTD - a non-validating processor can read a document that lacks a DTD but, as Sperberg-McQueen notes, this does not mean that it is wise to prepare documents in such a manner ("XML and What It Will Mean for Libraries"). XML documents without a DTD fail to take full advantage of the specialization possible with XML.

## Validity and "Well-formedness"

XML tags are formally defined in the DTD so that documents can be validated. If the XML specification is followed, the "parser can infer the structure of your documents from clues in the use of the tags" (Boeri and Hensel). However, unlike HTML, where nonstandard extensions or misspelled tags may be ignored by the browser, XML is strict in its adherence to the specification. This required error reporting should result in more consistent results among different processors. Throughout the literature, there are two major concepts underlying proper XML authoring. Documents must be "well-formed" and "valid." Well-formed documents are those that conform to the basic XML format. This means that elements must be nested properly with no overlapping tags. This is similar to a valid HTML document, where the first tag open must be the last tag closed, opening tags must have a corresponding closing tag, attributes must be enclosed in quotation marks, etc. At the most basic level, well-formedness simply requires that the document be created with proper attention to the characteristics of XML style. Validity requires that the document is well-formed and has (and follows) a DTD or else "starts with a Standalone Document Declaration" (Halbert). Exner and Turner define the well-formed constraints as "items that are absolutely mandatory under any circumstances," and the validity constraints as those that have to be followed "only under certain circumstances particularly if you expect that the document will be read with a validating parser" ("Examining XML"). An XML parser is software that checks for the validity and well-formedness of an XML document; it is similar in function to an HTML validator. However, unlike HTML, an XML document that is not well-formed will result in a fatal error and the document cannot be displayed. While XML documents must be well-formed in order to be displayed, not all XML documents must be valid. They can be viewed with a non-validating processor, or may result in a non-fatal error from a validating parser. Validity and well-formedness constraints are addressed in great detail in the XML specification because they are essential to the standard gaining widespread acceptance and remaining unproprietary.

## HTML and XML

XML is not intended to replace HTML; instead, the two are complementary. The two standards emphasize different elements of document preparation; HTML handles structure and presentation of text data while XML addresses "data semantics and meaning, as well as data presentation issues not addressed in a purely text-oriented language like HTML" (Exner and Turner). The World Wide Consortium echoes this view, emphasizing that XML does not replace HTML; in their Frequently Asked Questions, W3C states that "XML itself does not replace HTML: instead, it provides an alternative by allowing you to define your own set of markup elements." XML is suited for documents that require specialized data formats but, as Halbert notes, although XML allows you to "go beyond HTML . . you may not need anything but HTML for general purpose information distribution" ("Beyond HTML"). Experts predict that, due to the simplicity of HTML, it will "likely remain the language of choice for

encoding text and graphics as well as for arranging basic layouts" (Exner and Turner). XML will not immediately eradicate the popularity of HTML; instead, its immediate impact will be on specialized information and it will slowly spread from this arena to more widespread application.

As it is anticipated that many HTML documents may eventually be converted to XML, concerns have arisen about how to most effectively prepare HTML documents for the eventual conversion. While, as discussed above, the structure of HTML and XML tags is similar, differences include case sensitivity, mandatory inclusion of quotation marks, and the pairing of non-empty tags in XML (Exner and Turner). Since XML is still in the development stage, there are not currently any guidelines that specify the best way to prepare HTML for compatibility with XML. The only advice is to be consistent in the preparation of HTML documents; use a consistent case in tags, use quotation marks when specifying attributes even if HTML browsers do not insist on it, and close tags properly. Other changes, which will have to be incorporated in order to convert HTML documents to XML are not currently supported by Web browsers and so cannot yet be implemented. It seems likely that converting HTML documents to XML will be relatively straightforward; Exner and Turner suggest that either traditional Web browsers will be modified to read XML, or plug-ins will be available that will open an XML browser when necessary. Ultimately, there is still some uncertainty about the conversion process but, given the vast number of HTML documents that may eventually be converted, the process will not be difficult.

## Potential Applications of XML

Applications of XML are likely to be widespread and diverse. Obviously, XML's first major impact will undoubtedly be on the World Wide Web. XML will alter the way that Web pages are coded, the way Web searches are conducted, and it should benefit the efficiency of these searches. However, it is anticipated that XML's impact will also spread beyond Web applications to structured off-Web data as well (Boeri and Hensel). As XML becomes integrated with Web browsers, and Web browsers increasingly become the interface to information, XML's impact will extend far beyond the WWW. Because XML allows users to define their own tags in order to most effectively represent their data, this standard "provides the ability to formulate precise tags relating to librarianship or any other subject" (Exner and Turner). XML-based markup languages for subjects like mathematics and chemistry are already being developed. XML will facilitate a reprieve from the limitations of HTML. Graphs and diagrams cannot be displayed in text using HTML; instead, they require the inclusion of an image. These images cannot be readily manipulated and transferred to other applications. Similarly, as Exner and Turner note, complex science and mathematical formulas cannot be readily displayed; pages with multiple equations must often be stored and displayed in Portable Document Format (PDF), a form which also has limitations. Other data forms are also limited in how they can be displayed and received over the World Wide Web. HTML provides limited functionality with text and images, but does not allow for "interactive use of the many types of data" (Exner and Turner). XML will allow a greater variety of data to be transmitted and more effectively managed over the World Wide Web and in other applications.

## Benefits of XML

The benefits of XML, particularly in regards to the management of specialist information, are clear. XML allows individuals or groups of user to define their own document types and markup tags, opening this information to new uses. XML was designed by the World Wide Web Consortium, a non-proprietary collection of users; the result should be a product that reflects user needs rather than the marketing strategies of browser vendors. XML also advocates for documents that are more "well-formed" and "valid" than is characteristic of HTML documents, thus curbing some of the standardization issues that have arisen with current Web browsers.

## Drawbacks of XML

Undoubtedly though, the many advantages of XML are at this time countered by some disadvantages. Most of these disadvantages stem from the newness of XML. The most popular and widely available commercial browsers do not currently support XML, although it does seem likely that browsers like Microsoft's Internet Explorer and Netscape Navigator will soon support the specification. Another problem is that XML by itself does not have any presentation features; this requires additional XML-specific software like style sheets in eXtensible Style Language. However, it does seem that as XML gains wider acceptance, many of these problems will be reduced.

## Current State of XML

XML has been gaining increasing recognition since the World Wide Web Consortium's recommendation of the specification in February 1998. While there has yet to be the widespread development of products that support XML as has been the case with HTML, some tools are available for authoring and reading XML documents. An ever-increasing list of these tools can be found in The Web Developer's Virtual Library - XML Software Guide. The Guide lists browsers, editors, database tools, and other applications that support XML. Microsoft has also demonstrated interest in the new standard as is evident in their Web page XML Scenarios, which indicates that many of Microsoft's products, including Word and Excel will soon take advantage of XML's potential.

## The Future of XML

Experts have differing opinions about the future of XML. While many maintain that HTML will remain in widespread use for all but specialized information needs others, like Sperberg-McQueen assert that "HTML will become unimportant, a legacy format" or else an "attempt at a general-purpose DTD" ("XML and What It Will Mean for Libraries"). Most experts agree that DTDs will rise in importance, as will style sheets. They also agree that there will be attempts to privatize XML and make it proprietary and profitable. Only with the public input that has characterized XML's development to this point will the standard most effectively develop to serve the needs of users whose data representation requirements extend beyond HTML.

Top

---

# XML and Libraries

## Introduction

According to Halbert, XML is needed when "groups of users have data needs that go beyond HTML, which can usefully be specified in SGML-style DTDs" ("Beyond HTML"). Many libraries meet these criteria, as is evident in the increasing number of libraries currently employing SGML. The North Carolina State University Library SGML/XML Web page provides links to 87 Academic, 18 Government, and 52 General projects currently using SGML. A quick look at some of these sites confirms that many of the projects (particularly academic ones) are at least partially administered by the institution's library. Thus, it seems that SGML does, to some extent at least, meet the needs of libraries. As XML offers increased functionality over HTML but with less complexity than SGML, it seems that XML may pose a very real alternative for many libraries.

It seems likely that the first area in which libraries will notice and benefit from XML will be in their management of information resources on the World Wide Web. Libraries seldom have the money or are prepared to take the risks involved with a very new technology. Thus, it seems that many librarians' first introduction to the new technology will be accessing documents in XML over the World Wide Web. Exner and Turner suggest that "first of all, the advent of XML should cause the variety of materials available on the World Wide Web to increase" ("Examining XML"). They predict that the breadth of research information available over the Web will increase as specifications for new fields are created. Library patrons have learned to rely on the Web as an information source and libraries, if they are to continue to facilitate this access, must understand the developments in information availability from this source. XML introduces the possibility of "structured information management on the WWW" (Halbert) and it seems likely that libraries will first respond to XML in this context. Exner and Turner predict that XML will have an enormous impact on the publishing industry; XML will have greater versatility than PDF, and will be more cost-effective and less labour-intensive to implement that SGML. They predict the result will be substantial changes in the field of electronic collections. Library web masters will also face to challenges as they try to keep digital collections up to date; XML is undoubtedly more labour-intensive than HTML, but its added functionality means that electronic collections may become more prominent in libraries. Exner and Turner cite other ways in which XML may revolutionize the library, including electronic reserves which will "see tangential effect, as professors can put greatly varied materials on the Web," and the "ever-increasing field of distance education" which may also see effects from "the new variety of data potentially coming to the Web with XML"(Exner and Turner).

## Databases

One of XML's most useful applications in libraries may be its potential in database production. Emmott suggests that it is "here that XML stands out from SGML" because it is "flexible, portable and accessible to newcomers - thereby creating the wider context." XML escapes the limitations of the "page horizon" (Emmott) because while HTML operates in units of one or more pages, XML can operate in units of any size, from one character all the way up to entire collections. Because XML can work "in terms of fragments of content" (Emmott), the databases managing this content must be more flexible than traditional Relational Database Management Systems. Instead, Object-oriented Database Management

Systems are required: XML is placed "at the centre of the process - [a] repository of fragments. From it documents are produced: a paper document, an on-screen presentation, a web-site, a CD-ROM, etc." (Emmott). The importance of the document is diminished in favour of the information it contains. Content is reshaped so that relevant fragments, rather than the extraneous material inherent in the document as a whole, are retrieved. XML has the potential to revolutionize databases as we know them, not just by changing the markup language used to encode the data, but by changing the way we conceptualize and approach information retrieval.

XML also has implications for the portability of database information. XML tags can be customized to allow the author to add a higher level of information or metadata to the document. Metadata can provide "commentary or qualifications for the encoded data, including semantics and meaning" (Exner and Turner). Thus, in the case of a MARC record, field information could be embedded directly into each specific tag. Each tag contains both the field information and the data. This information could then be imported directly into a database or OPAC. The tags themselves instruct the program how to interpret the field names and the data contained in each field. The result is records that are more portable and that can be manipulated much easier that HTML currently permits. XML could eliminate the need for MARC tags and other specialized codes that instruct programs how to interpret data; many standards could be replaced by one standard (XML) which would reduce the need for specific field tags and ultimately make data transferable between previously incompatible systems.

## Other Potential Applications

XML introduces several opportunities for libraries. Sperberg-McQueen identifies some of these, including new and more capable software developments, the delivery of information rather than just pages, smart pages, database exchanges, smart applications, and work-flow changes ("XML and What It Will Mean for Libraries"). However, these increased opportunities will also mean increased responsibilities for librarians and information professionals. Effective use of XML requires expert knowledge of the subject-area being represented, and extensive content analysis to ensure that the appropriate DTDs and corresponding tags are selected. As Sperberg-McQueen notes, "XML is not an out-of-the-box solution" ("XML and What It Will Mean for Libraries"). Application of XML still requires DTDs, style sheets, usage conventions, and processing software. The newness of XML is also currently a disadvantage to the adoption of the technology in libraries. The cost of the new software may be prohibitive to both libraries and users. As well, the XML specification is very new since the XML 1.0 standard recommendations were just issued by W3C in 1998. The newness of the standard raises questions about how well-received and widely adopted it will become. The standard has met with enthusiasm from the Internet community, but it seems likely that it may pass through more revisions and changes before it settles (as much as any Internet standard can settle) into a widely adopted form.

## XML Projects of Interest to Librarians

While SGML is currently being used in a number of library and information contexts (see "XML at NCSU" for a list of SGML projects), XML is not yet in widespread use in library and information environments. There are, however, a few projects being undertaken that do employ the new technology. One such project is MD Consult, a collaborative venture of publishers Lippincott Wiliams & Wilkins, Mosby, W.B. Saunders, and others. While MD Consult is not a library-produced product, it is a database

that employs XML and is being marketed to libraries. XML is particularly suited to this kind of use because it can be customized to the specific needs and terminology of different fields, such as medicine. MD Consult is designed for access via the World Wide Web and allows doctors "to search across multiple databases and combine disparate search elements effectively" (Saunders). For example, the XML technology underpinning the database allows patrons to not only enter a drug name, but to limit their search to "drug-prescribing information, therapy costs, and interactions" (Saunders) rather than entire articles. MD Consult is an early example of the new databases that are made possible by the adoption of XML.

A few libraries and information centers have also tentatively begun to explore the potential of XML. Two examples are the North Carolina State University Libraries Special Collections Department and U.S. National Information Center for Educational Media (NICEM). North Carolina State University Libraries Special Collections is currently undertaking a large document description project similar to those being carried out by a number of academic libraries. The project uses EAD (Encoded Archival Description), the DTD for archival finding aids. Right now, the project is using the beta version of EAD, but version 1.0 of EAD, which was released in 1998, is XML compatible and the project will be converting to version 1.0 soon. It is anticipated that the conversion should go smoothly because version 1.0 of EAD does not use many of the complex features of SGML (Kiel). Another example of XML implementation is The National Center for Educational Media , which converted its database from "a simple relational database management system to an XML-based intranet system" ("NICEM Implements"). Bibliographic records are stored as XML documents; thus, the metadata inherent in the tags describes the educational audiovisual material. The bibliographic records can be used by any system that meets standardized criteria. Ultimately, the aim of both of these projects is employ XML to facilitate more advanced information retrieval, where searching will result in "more logical retrieval sets" ("XML at NCSU").

The potential of XML to revolutionize libraries is confirmed in OCLC's current Mantis project. Mantis is a "toolkit for building Web-based cataloging systems with arbitrary metadata definitions and interfaces" (Schafer). The Mantis project may provide some insight into the future of XML applications in libraries. The user benefits from the added functionality of XML but is shielded from its complexity. As Schafer notes:

> By requiring that end users have only standard Web browsers, the project is aimed at lowering the barriers to acceptance, distribution and use of these tools. By emphasizing that metadata can assume many different formats, the project shows that generalized metadata tools can be used for many applications.

Schafer describes the system by providing an example of its use. A user searches a metadata database and wants to display a record. The metadata record must then be reformatted so that the user can view or edit it. Since Mantis stores arbitrary metadata records, the system must be told about the record in question so that it knows how to format it. This is where XML comes in. An XML template contains information about how to display the record. The system "takes the record out of the database and populates the XML template with data from the record to create a populated XML template" (Schafer). The system is combining the abstract data model (the XML template) and data from the concrete record. The populated template is then transformed for the user. All data and display instructions in the populated XML template are translated into standard HTML, which is sent to the user's browsers. The

process also works in reverse. Authorized persons can edit a displayed record using standard HTML because "the display process embeds hidden information about the data fields into the HTML it produces so that standard HTML forms can be used to edit the record" (Schafer). Once the editing is complete, the record is submitted to the database; upon submission "the populated XML template is compared to the original XML template to extract the data to be stored in the database" (Schafer). Thus, Mantis users benefit from the added functionality of XML when searching the database, but are shielded from its complexity when searching and editing records. The Mantis project is evidence that OCLC takes XML's potential in libraries seriously, and also indicates the direction that implementation of XML in libraries may take.

Halbert's discussion of the SAGE Project (Selected Archives of Georgie Tech and Emory digital archive project) exposes some of the difficulties faced by library and information environments currently trying to implement XML. The emerging nature of the XML standard has made its adoption and implementation problematic. The current lack of XML browser applications has left the project with two options: either wait for a standard XML browser to emerge, or develop a browser suitable to the needs of the project. The first option is problematic because it is unknown when or if this will happen. The second option also poses problems, because development of a browser would be costly and complicated. Ultimately, the project has chosen the route of many such projects; the project is being completed in SGML with plans to eventually transform it to XML. Halbert's article is valuable because it identifies some key technical issues for the future. First, he points out that libraries have specialized needs for Web data that extend beyond the capabilities of HTML which could be addressed by XML. Secondly, he notes that in order to realize the benefits of XML, libraries must first standardize XML DTDs for relevant data types, and then either develop these DTDs themselves or purchase them. His final comment is that cooperative effort is most likely to achieve these goals. The trend towards library consortia may take a whole new importance in light of the need for XML developments. He cites projects like the Digital Library Federation's XML in Digital Libraries initative, as a reason for optimism regarding the future of XML in library and information environments.

Top

---

# Conclusion

XML looms on the horizon but the truth about the role it will play in digital representation is not yet known, and its potential impact on library and information environments remains just that, potential. Yet, the relationship between XML and these information climates seems promising; as Exner and Turner note, "XML is certainly a significant advance in the handling of data and information in the Web environment, and anything that affects information will also impact the library field" ("Examining XML"). Librarians are well-advised to be aware of technological developments that may have a profound impact on the way they manage and deliver information. XML is one such technology deserving of attention.

Top

---

# Bibliography

## XML in Library/Information Environments

"A Leaner, Meaner Markup Language." *Online & CDROM Review* 21 (June 1997): 181-4.

Burrows, Toby. "Beyond HTML: Markup Languages and the Future of Electronic Information." *Australian Academic & Research Libraries* 29.2 (June 1998): 150-6.

Davies, Ron. "XML: A New Web for the New Century." *Feliciter* 44.10 (Oct. 1998).

Digital Library Federation. http://www.clir.org/diglib/dlfinit.htm. (1 March 1999).

Emmott, Stephen. "At the Event: SGML, XML, and Databases." 10 Dec. 1998. http://www.ariadne.ac.uk/issue18/emmott-sgml/ (2 Feb. 1999).

Exner, Nina, and Linda F. Turner. "Examining XML: New Concepts and Possibilities in Web Authoring." *Computers in Libraries* 18.10 (Nov/Dec 1998). http://www.infotoday.com/cilmag/nov/story2.htm (2 Feb. 1999).

Halbert, Martin. "Beyond HTML: The EAD and XML Formats, and the Development of Digital Archives: A Presentation Given at the 1998 ALA Annual Conference." 26 June 1998. http://elan.library.emory.edu/Misc/Conf/IRIG-XML/ (7 Feb. 1999).

Hockey, Susan. "Describing Electronic Texts: The Text Encoding Initiative and SMGL." http://lcweb.loc.gov/catdir/semdigdocs/hockey.html (2 Feb. 1999).

Kiel, Paul. "XML." 2 March 1999. Personal e-mail. (2 March 1999).

MD Consult. http://www.mdconsult.com. (1 March 1999).

National Center for Education Media. http://www.nicem.com/nicefram.htm. (1 March 1999).

"NICEM Implements Two New Database Applications to Boost Quality of Records." *Computers in Libraries* 18.10 (Nov/Dec98):16.

Sage Project. http://sage.library.emory.edu. (1 March 1999).

Saunders, Laverne. "Not Your Mother's HTML: Moving on to XML." *Computers in Libraries* 18.10 (Nov/Dec98): 45.

Keith Shafer. "Mantis project provides a toolkit for cataloging." *OCLC Newsletter* 236 (Nov/Dec 1998). http://www.oclc.org/oclc/new/n236/research_mantis_project.htm (15 Feb. 1999).

Sperberg-McQueen, C.M. "XML and the Future of Digital Libraries." *Journal of Academic Librarianship* 24.4 (July 1998): 314-17.

Sperberg-McQueen, C.M. "XML and What it Will Mean for Libraries." 30 June 1998. http://www.uic.edu/~cmsmcq/talks/teidlf1.html (2 Feb. 1999).

"TEI and XML in Digital Libraries." 30 June - 1 July 1998. http://www.hti.umich.edu/misc/ssp/workshops/teidlf/ (7 Feb. 1999).

"XML at NCSU Libraries Special Collections." North Carolina State University Libraries. 30 July 1998. http://www.lib.ncsu.edu/libraries/archives/tech_serv/ead.html (7 Feb. 1999).

## XML - General

Berners-Lee. "The World Wide Web: Past, Present and Future." August 1996. http://www.w3.org/People/Berners-Lee-Bio.html/1996/ppf.html (14 Feb. 1998).

Boeri, Robert J., and Martin Hensel. "XML: The New Document Standard." *EMedia Professional* 11.6 (June 1998): 33.

Bray, Tim. "Stretching the Document Concept" *Web Techniques* 3.12 (December 1998): 43-46 http://www.webtechniques.com/features/1998/12/bray/bray.shtml (2 Feb. 1999).

Dan Connolly. "The W3C XML Extensible Markup Language Working Group Home Page." 2 Feb. 1999. http://www.w3.org/XML/ (7 Feb. 1999).

Cover, Robin. "The SGML/XML Web Page." 2 Feb. 1999. http://www.oasis-open.org/cover/sgml-xml.html (2 Feb. 1999).

Dudrow, Andrea & Joanna Pearlstein. "XML Format May Fortify Web Content." *MacWeek* 11.42 (31 October 1997). http://macweek.zdnet.com/mw_1142/nw_xml.html (7 Feb. 1999).

Kelly, Brian. "What Are . . . .Xlink and Xpointer?" *Ariadne* 16 (July 1998). http://www.ariadne.ac.uk/issue16/what-is/intro.html (7 Feb. 1999).

Kelly, Brian. What is XML? *Ariadne* 15 (May 1998). http://www.ariadne.ac.uk/issue15/what-is/ (7 Feb. 1999).

Mace, Scott, et.al. "Weaving a Better Web." *BYTE* 23.3 (March 1998):58-68. http://www.byte.com/art/9803/sec5/art1.htm (7 Feb. 1999).

Microsoft Corporation. "XML Scenarios." 11 August 1998. http://www.microsoft.com/xml/scenario/intro.asp. (1 March 1999).

Pepper, Steve. "SGML & XML Tools - By Vendor Name." 11 Feb. 1998. http://www.lib.ncsu.edu/libraries/archives/tech_serv/sgmldemo.html (14 Feb. 1999).

Powell, Thomas A. "Extend the Web: An XML Primer." *Internet Week* 691 (24 November 1997): 47-49. http://www.techweb.com/se/directlink.cgi?INW19971124S0090 (7 Feb. 1999).

Radosevich, Lynda. "XML Initiatives Take Shape." *InfoWorld* 19.37 (15 September 1997). http://www.infoworld.com/cgi-bin/displayArchive.pl?/97/37/t02-37.1.htm (7 Feb. 1999).

Radosevich, Lynda, and Dana Gardner. "XML Gets Even Closer to the Web." *InfoWorld* 19.50 (15 Dec. 1997). http://www.infoworld.com/cgi-bin/displayArchive.pl?/97/50/t07-50.6.htm (7 Feb. 1999).

Sall, Ken. "XML Software Guide." 1 March 1999. http://wdvl.com/Software/XML/. (1 March 1999).

Willner, Eli. "Preparing Data for the Web with SGML/XML." *Information Today* 15.5 (May 1998): 54.

Willner, Eli. "Preparing Data with SGML/XML: Part II." *Information Today* 15.6 (June 1998): 52-4.

Willner, Eli. "Preparing Web Data with XML: Part III." *Information Today* 15.7 (July/August 1998): 39+.

"XML for Managers: Evaluating SGML vs. XML from a Manager's Perspective." ArborText White Paper. http://www.arbortext.com/Think_Tank/XML__Resources/XML_for_Managers/xml_for_managers.html (7 Feb. 1999).

---

Top

**This document was last revised on March 17, 1999.**
**e-mail comments to: csb1@gpu.srv.ualberta.ca**