XML: What, What, Who and Where.

Edward A. Boyno, Ph.D.
Dept. of Computer Science
Montclair State University
Montclair NJ, 07043
boynoe@mail.montclair.edu

Abstract

When people speak of XML, the eXtensible Markup Language, they are usually referring not only to XML itself, but to an array of tools and technologies that use XML or make it more useful. Such things as XSL, XPath, XQuery, FLWOR, SOAP, AJAX and others are all part of the "XML-Complex." After a brief discussion of what the XML Complex is and what it is used for, the paper will discuss who ought to be learning about it and where in the curriculum it ought to be taught.

Keywords: Curriculum, XML, Ajax, FLWOR, SOAP, XPath, XSL.

1.0 Introduction.

The paper attempts to show the importance of the XML complex in all disciplines that deal with document processing. It will argue that the XML complex belongs in every Information Technology/Computer Science curriculum, indeed that it belongs in the curriculum for all information specialists including Library Science.

Sections 2 and 3 of the paper present a very brief introduction to the more prominent members of the XML complex, including XML itself. Section 2 discusses some of the technologies that are used to control or modify XML documents and section 3 presents some applications that use XML documents to perform useful services. In section 4, the paper presents a list of some of the subject areas and applications in which the XML complex has become indispensable. Section 5 makes suggestions about where and when the XML complex might belong in a curriculum and section 6 summarizes the other sections.

2. What is XML and what are the applications that act on XML documents.

2.1 XML

A markup language is a set of words and symbols for describing the identity of pieces of a document. (Flynn, 2006). Using a mark-up language, one can, for example, specify that one part of the document is a heading, while another part is a paragraph and so forth. Some mark-up languages, notably the "Hypertext Mark-up Language" (HTML) also have codes, which contain formatting information.

XML is a mark-up language. It is "extensible" in that it allows users to create their own codes, called tags, which, as in HTML, come in pairs and act as parentheses. User defined tags allow the addition of semantic content to an XML document in a way that is impossible in HTML.

A set of tags together with whatever comes between them is called an element. Elements may contain text or other elements and may, optionally, contain a set of attributes. An XML document is then correctly described as a tree of nested elements in

which the children of a node are explicitly ordered. An example (The first tag identifies it as an XML document):

```
<?xml version = "1.0" ?>
<order number = "312597">
<!- - number is an attribute-->
   <date>11/7/2006</date>
   <customer id="1234">Foo Bar Industries</customer>
   <item color = "Blue">
      <part-number>E61</part-number>
      <quantity>16</quantity>
   </item>
   <item color = "Red">
      <part-number>MA34</part-number>
      <quantity>1</quantity>
   </item>
</order>
```

2.2 Technologies that act on XML Documents.

This section presents some of the more common of these applications. They are listed in no particular order since there is not really any natural linear ordering.

2.2.1 Application Interfaces. These tools provide a set of methods that can be used by an application, such as a browser, to analyze an XML document, place it into a well-defined tree structure and to extract information from it. Two commonly used application interfaces are the Document Object Model (DOM) and the Simple API for XML (SAX). The DOM requires the entire XML document to be parsed and in memory It is thus suitable for operating on a document that must be accessed non-sequentially or repeatedly. SAX is an on demand processor which is more suitable to documents which are operated on sequentially and/or whose elements are needed only once.

2.2.2 Schema Definition. One of the great strengths of XML documents is that they are self describing. It is often desirable however to impose a uniform schema for an XML document or set of XML documents. Both of the following technologies provide that functionality.

2.2.2.1.Document Type Definition (DTD). A DTD is a regular expression which provides a schema for XML documents. A sample DTD (for the previous XML example):

```
<?xml version ="1.0" ?>
<!ELEMENT order ( date, customer, (item)+)>
<!ATTLIST number CDATA>
<!ELEMENT date (#PCDATA)>
<!ELEMENT customer( #PCDATA)>
<!ELEMENT item (part-number, quantity)>
<!ELEMENT part-number( #PCDATA)>
<!ELEMENT quantity (#PCDATA)>
```

Many specialized DTD's for domain specific XML documents have been created including MathML and ChemicalML.

2.2.2.2 XMLSchema XML Schema is another, more recent, more advanced technology which is also meant to allow users to describe the schema of an XML document. XML Schemas are themselves XML documents. The DTD schema above as an XML schema:

```
<?xml version="1.0">
<schema>
<element name="order">
 <complexType>
   <sequence>
     <element name="date" type="string" />
      <element name="customer" type="string" />
      <element name="item">
        <complexType>
          <element name="part-number' type="string" />
           <element name="quantity" type ="string" />
        </complexType>
      </element>
   </sequence>
 </complexType>
</element>
</schema>
```

2.2.3 XPath XPath is a language which is used by both XSL and XQuery (below) to specify a node in an XML document. An XPath instruction is very like a path specification in Unix with the significant difference that a step in the path may specify the ordinality of a child and/or contain a filter. XPath instructions, by definition, return a set of nodes even if that set contains only one node. Three XPath examples:

/order/item[2]/part-number

/order/item[@color='Blue']/quantity

../part-number

The first two lines are "absolute" paths that begin at the root element. The first specifies the ordinality of an item element and therefore returns the part-number element of the second item child of an order element. The second contains a filter on the color attribute of the item element and so returns the quantity element of any items whose color attribute is 'Blue'.

The third example contains a relative address. It is presumed to start at a quantity node and returns the part-number element of its parent.

2.2.4 Document Transformations. Both of the tools discussed in this paragraph are used to transform XML documents. .They can be used to change an XML document or to search a document for a particular piece of information, that is, act as query language.

2.2.4.1 eXtensible Stylesheet Language (XSL). XSL is a declarative programming language, that is, it doesn't require the programmer to specify how something is to be done, only what the programmer wants to be done. In the case of XSL, actions are taken when specified nodes in an XML document are encountered. An XSL Transformation (XSLT) takes as input an XML document and returns another document. This returned document could be another XML document, an HTML document or pretty much any other kind of document. XSLT's are themselves XML documents and can therefore be transformed by other XSLT's.

XSL is a rule based language. It looks for various nodes in an XML document and when it finds one that it recognizes, it fires a rule which produces some output. The basic unit of an XSLT stylesheet is a template which has the syntax…

<xsl:template match={match pattern} >

where the match pattern is an XPath expression.
An example:

<xsl:template match="/order/item">

Which sets the "context" node to be the first item element child of the order element.

The "apply template" template tells XSL to suspend operating and to process a set of nodes, as in
<xsl:apply-templates select = "item">

which tells XSL to process any templates which apply to elements of type item. A complete example:

```
<?xml version="1.0"?>
<xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<!--Finds the quantity of items whose part-number is E61-->
<xsl:template match="/">
<p style="font-size:120%">The number of items with part-number E61 that were ordered is:</p>
<xsl:apply-templates/>
</xsl:template>
<xsl:template match="order/item/part-number[.='E61']">
<!—Prints out the value of the quantity element-->
<p style="font-size:120%"><xsl:value-of select ="../quantity" /></p>
</xsl:template>
<xsl:template match="text() | @*">
<!--override the default templates and do nothing for everything else-->
</xsl:template>
</xsl:stylesheet>
```

As noted in the comment, this XSLT processes the sample XML document and returns the number of items whose part number is E61

2.2.4.2 XQuery. XQuery can also be used to extract information from XML documents, generate summary information from them and transform them. Navigation is accomplished by the use of expressions that are 99% the same as XPath. It uses a procedural language called FLWOR because of its five basic instructions: "For", "Let", "With", "Order By" and "Return". It can op-

erate on multiple documents which among other things allows it to perform joins. An XQuery example which uses FLWOR to perform a join on two hypothetical documents.

```
for $proj in doc("projects.xml")/Projects/Project
for $emp in doc("employees.xml")//Employee
where $proj/@owner = $emp/@id
return $proj/name, $emp/name
```

3. Applications that use XML documents

3.1 SOAP. the Simple Object Access Protocol. SOAP is a means for an object on one computer to make use of objects on other computers. It is the foundation layer of the "Web services" stack, providing services that higher layers can build upon. XML is used to represent the information being transmitted.

3.2 Ajax. Ajax is an acronym whose letters stand for Asynchronous Javascript and XML. It uses a variety of technologies such as XHTML (Extensible Hypertext Markup Language), DOM, JavaScript and/or SOAP to allow web browsers to asynchronously exchange small bits of information with a server instead of having to constantly reload web pages. (W3 schools Ajax tutorial) XML is the format used in the data transfer.

3.3 RDF, the Resource Description Framework. RDF is a framework for describing web resources such as the title, author and modification date of a web page. It is written in an XML language called, appropriately enough, RDF/XML. It provides a common way to describe information for applications and because it is written in XML can be used by different systems and languages. It can be used in such wildly divergent areas as the description of the price and availability of shopping items or the description of digital libraries.

3.4 Database Management. XML is not in and of itself a database management system (DBMS). Among other things it totally lacks any transaction management capabilities. Since XML is a such a versatile tool for data storage, however, many attempts have been made to embed it into a true DBMS. These databases come in two flavors:

XML enabled databases. Oracle, DB2, Sybase and others now support XML documents as a native data type. A variant of SQL, SQL/XML is used as a query language.

Native XML databases: The basic unit of storage in these DBMS's is an XML document. Mark Logic among others markets such a DBMS. It uses XQuery as a query language.

4. What is the XML complex used for?

There are two, not necessarily independent, answers. The first is simply data transmission. XML is an excellent vehicle for data transfer. It is very useful in moving data among web sites and can be used when data must be gathered from multiple, heterogeneous sources. The second, which perhaps includes the first as a special case is "Document processing". A quick and dirty search of the internet and the literature, especially the XML conferences since 2001 yielded the following, non-database, subject areas in which the XML complex has become an essential tool:

4.1 CAD/CAM and Graphics. An XML document can be used to hold graphical data. This means that graphics need no longer be static objects simply embedded in text. but, can be represented in semantically meaningful units. These units can then be assembled into composite graphics. (Gaudron 1999),(Williams, 2004)

4.2 Chemistry/Biochemistry. Murray-Rust and Rzepa (Murray-Rust, 2002) argue "that XML offers a general powerful and extensible mechanism for handling both the capture and the publication of chemical information, and most particularly for the first time will allow this process to operate equally well in both directions."

4.3 Document Management

4.3.1.Contracts. There is a great and growing need for the automated handling of contracts. Contract enforcement and the tracking of litigation involving contracts are two of the issues which practitioners are addressing. The natural solution for representing a contract is XML and a standards group has been working on the problem

since 2001. More detailed discussion can be found in (Meyer, 2005) and (Winn, 2005).

4.3.2. Library services. One can do no better than to quote E.L. Morgan (2004) from his tremendously useful "Getting Started with XML". A manual originally written for librarians and other information specialists:
"..libraries are becoming less about books and more about the ideas and concepts manifested in the books. (In this sphere of influence) there needs to be a way to move data and information around efficiently and effectively. XML data shared between computers and computer programs via the hypertext transfer protocol represents an evolving method to facilitate this sharing."

4.3.3. Medical records. The FDA has mandated that all drug product descriptions be in XML format. Standards are also being developed for medical records in general and for prescriptions. (Waldt, 2005)

4.3.4. Publishing. From on-line publishing of course catalogs (Cummings, 2005) to the ability of publishers to let their clients effectively design individualized copies of books, (Hunter, 2005) XML pervades the publishing industry.

4.3.5. Political Science. In their case study presented at XML Conference & Exposition, Richards and Hatter (2005) describe the experience of the Parliament of Ireland, among other legislative bodies, in the use of XML for collaborative authoring of structured documents.

4.4 E-Commerce. Among dozens of other things, XML can be used to validate invoices (Brun, 2005), to produce web forms (Boyer, 2005) and to manage metadata(Greenfield, 2004). XML is nearly ubiquitous in E-Commerce.

4.5. Financial Services. It suffices to quote Mark O'Neill, (2005): "XML is becoming the de facto business document interchange language."

4.6. Knowledge Management. Knowledge Management applications must deal with large amounts of often inconsistent data. They must be able to classify information into meaningful structures and to distribute information across an enterprise (Sperberg, 2004). XML based tools, especially RDF, are being used to support these basic Knowledge Management tasks.

4.7 Mathematics. Some areas of mathematics such as computational algebra have algorithms that, although they have larger than exponential complexity, are easy to distribute. Mathematicians often do not have access to sufficient computing resources and seek to take advantage of slack time on the computers of a given institution to form a computing grid. XML would be the data exchange language of the grid. (Milowski, 2005)

4.8. Military Science. XML is being used to represent virtual forces in military simulations. (Hobbs, 2003)

4.9. Multimedia. Traditional multimedia tools are ill equipped to handle some of the business realities for which their output is used. Sales brochures, for example, must synchronize the same content for both print and web pages. The tool of choice for such tasks is XML. (Severson, 2004)

4.10. Web Services. Broadly speaking a web service is an application that uses internet protocols to allow a client to exchange information with a server. XML documents can be used as a data store on the server and as a standard for communication between nodes. XSLT can be used as the engine that drives a web service. SOAP is an example of a web service.

Anyone who remains unconvinced of the broad application of the XML complex, can browse the proceedings of the 2005 XML Conference and Exposition at 2005.xmlconference.org/proceedings . The site also contains links to previous years proceedings.

5. Who needs to know XML?

Not a difficult question. It is evident that anyone who deals with documents of any kind needs at least a basic knowledge of the XML complex. Since just about every profession deals with documents in some way, one must conclude that just about everybody needs such a basic knowledge.

It is also evident that people who are in charge of document processing and/or handling need a more thorough knowledge of the subject. Included in this group are computer scientists, IT professionals and librarians.

Finally, any one who deals with databases, data processing or data transfer needs an extensive knowledge of XML. XML is also such a powerful tool for providing web services that any one who is studying those applications must also have a deep knowledge of XML. In short, the author believes that at least some knowledge of XML should be part of every student's college education.

6. Where should it be taught?

The author argues that XML should be taught as early as possible in all relevant curriculums. The author routinely includes a little XML in the freshman level, non-major course with some success. In that course, students must produce a 1500 word paper as part of the basic requirements of the course. They are instructed to markup their paper using a simple format and were provided with an XSLT that transforms their original document into an HTML document that they could see in their browser. Students are given a chance to make small changes in the XSLT and generally enjoy being able to play with the format of the final document.

The author believes that XSL and/or XQuery+FLWOR should be part of any introductory programming sequence (if only to provide contrast to whatever programming paradigm they're using in their class) and that XML tools should become a routine part of higher division courses, such as Software Engineering. Finally XML databases should appear in any database course

7. Conclusion

Every subject area which is the province of information technology involves document processing to a greater or lesser extent. The paper has attempted to show that every subject area that involves document processing uses the XML complex in some way. It argues that XML should be part of every IT curriculum and further that XML should be introduced as early as possible.

8. Works Cited

Asaravala, A. InfoWorld; 10/17/2005, Vol. 27 Issue 42, p22-28

Brun, M., Nielsen, B., Lanng, C and Rasmus sen, B. "Large Scale Validation of Millions of UBL invoices with XML Schema and Schematron". XML Conference & Exposition, Atlanta, GA. 2005.

Boyer, J. "Enterprise-Level Web Form Appli cations with XForms and XFDL" XML Conference & Exposition, Atlanta, GA. 2005.

Brundage, M. XQuery, The XML Query Lan guage. Addison-Wesley, Boston, 2004.

Cummings, D. "Higher Education: Course Catalogs" XML Conference & Exposition, Atlanta 2005.

Flynn, P. ed. /xml.silmaril.ie/basics/markup June 2006

Gaudron, M. "Producing and Using Intelligent Graphics in XML/SGML Electronic Publishing." XML Europe, Grenada 1999

Greenfield, J. "Models and Metadata: the Role of XML in Enterprise Development." XML Conference & Exposition, Washington D.C. 2004

Hobbs, R. "Using XML To Support Military Decision-Making" XML Conference & Exposition, Philadelphia, PA. 2003.

Hunter, D., Kirt Cagle, K., Dix, C., Kovack, R., Pinnock, J. and Rafter, J. Beginning XML. Wrox Press, Birmingham UK, 2001

Hunter, J. "XQuery by Example: Making O'Reilly Books Sing and Dance." XML Conference & Exposition, Atlanta 2005.

Meyer,P. "A proposed XML standard for contract documents" XML Conference & Exposition, Atlanta, GA. 2005.

Milowski, A.R. "Computing for the Mathe matical Sciences using XML, web services and P2P" XML Conference & Exposition, Atlanta 2005.

Morgan, E.L. "Getting Started with XML". infomotions.com/musings/getting-started 2004

Murray-Rust, P. and Rzepa, H. "XML in Chemistry www.ch.ic.ac.uk/rzepa/iupac/ (2002)

O'Neill, M. "Securing XML- case studies from the Financial Services Industry" XML Conference & Exposition, Atlanta 2005.

Richards, L. and Hatter, C. "Drafting Legisla tion in XML" XML Conference & Ex-position, Atlanta 2005.

Roberts, G. "Using Web Services and XML Harvesting to Achieve a Dynamic Web". Computers in Libraries; Jun2005, Vol. 25 Issue 6, p30-32

Severson, E. "XML for Creative Content and Page Layout Applications". XML Con-ference & Exposition, Washington DC 2004.

Sperberg, R. and Voleti, R. "Using RDF for knowledge management" XML Con-ference & Exposition, Washington DC 2004.

W3schools. RDF Tutorial. www.w3schools.com/rdf/rdf_intro.asp .

W3schools. Ajax Tutorial. www.w3schools.com/ajax/default.asp

XML Marks the Future for Electronic Records. Information Management Journal; Nov/Dec2005, Vol. 39 Issue 6, p64-68

Waldt, D. "XML Initiatives in Pharma" XML Conference & Exposition, Atlanta 2005.

Williams, J. "Applying Techniques of Textual Reuse to Graphics Using SVG and XML" XML Conference & Exposition, Washington DC 2004.

Winn, Jane. "The Impact of XML on Contract Law and Contract Litigation" XML Conference & Exposition, Atlanta 2005.