Educational Object Repositories and XML

Abstract
As research continues on educational object repositories there is a growing awareness that there is a lot of work required to deal with a huge amount of heterogeneity between institutions and departments. There is a lack of a common ground that will allow them to communicate and exchange information. XML (Extensible Mark-Up Language) is one technology that is evolving with some solutions to these problems. It can provide a number of options that will allow data mobility between repositories and systems. It also provides a raw format that will allow existing and developing software to manipulate and change the data to suit the needs of the client using the system. It may provide the universal language that can break down existing barriers to discovering and sharing educational content in a digital environment.

Introduction
Educational object repositories were designed to fulfill the needs of educators looking for quality digital educational content on the Internet. They were created to deal with the increasing difficulty in finding these resources on the Internet. This problem is due to the sheer number of websites and a lack of any standardized way of describing the content in those websites. This frustration has led a variety of educational institutions and their departments to develop their own solution to the problem. As a result each of these repository solutions is designed to meet the need of a particular institution. Each contains different content and educational focus that has resulted in a variety of ideas about how the repository should be organized and presented. Adding to the complexity, the decisions for implementation are often linked to budgets, functionality, security and local expertise. This mixture of solutions has resulted in a variety of data structures that range between flat file, relational and object models. All are suitable as a basis for an object repository as they all provide a solution that works for that particular institution. Unfortunately, they also create an even worse situation for the rest of the world still trying to discover and use online educational materials. When all the materials were located on the Internet, embedded in a standard web page, there was at least a chance that educators outside of the group of repository users might locate them. Many of the current solutions for discovering resources online, such as Internet search engines, at least had a chance to find and index them. Now they are hidden inside an asset management system that is more than likely not designed to communicate with other like repositories as well as the rest of the world.

So there is a problem if there is a need to move the educational object metadata beyond the repository environment. As the repository serves the students and staff of an educational institution there might not be an immediately apparent reason for ever trying to migrate the data outside of that functioning environment. The need arises when
different institutions decide to share their resources in order to provide the greatest amount of quality material to their faculty and students. This would require that the metadata about the educational material somehow become available to all those cooperating institutions.

Questions arise about how the data will be represented, how it will be allowed to move to a new repository and how will that new repository view the metadata? Fortunately, there are a number of other groups using the Internet that are dealing with similar problems of heterogeneous repository systems and the metadata used to index and describe their content. One solution to dealing with this kind of problem is XML (Extensible Mark-Up Language) (WC3 2000). XML is a markup language for documents containing structured information. It allows the data to be viewed, manipulated and even moved into other databases. It provides a degree of flexibility that will allow data to be converted into whatever form is necessary to make it useful. It provides the hope that educational institutions will be able to freely share information about their educational content.

Data Mobility

One reason to convert the metadata records contained within an educational object repository into XML is mobility. XML files can be used to backup the contents of a metadata repository or allow the movement of data between databases. It requires that the metadata data in the system have some kind of processor that can take the raw database information and convert it into XML. This often involves writing code that can create XML from the database.

The XML format provides a universally recognized construct of information. Once it is converted into this format it can be interpreted by any system that has been given the schema or structure of the XML document. The WC3 has developed XSLT to deal with the transformation of XML from one format to another (WC3 1999). It allows you to transform a proprietary data structure into a form that a different system can understand. This is one of many different technologies that can transform the XML into another format that can then be imported into another system. The drawback to such a system is that a transformation is required for each different type of XML file that is output from the different systems (figure 1).

To solve this you can move metadata out of the system into a XML “hub” format that can be used by all systems (figure 2). This is a universal, standard format for moving data between systems. This means that only one filter or transformation processor has to be written for each system rather than a different one for every system that wants to share data (figure 2).
It is important to realize that it is still possible to lose data using both systems. If information is contained in the XML file that has no place to go in the target system it will likely be lost. This might include administrative metadata that was only relevant to a particular institution or discipline specific metadata that is of no use to a general audience. The actual structure of the information can also be affected. As the XML schema become more complex it begins to generate problems when the data is moved back into a relational database structure (Shanmugasundaram et al. 1999). Relationships defined in the XML data moving into the relational model can provide a challenge in both performance and complexity. It may involve sacrifices in the fidelity of the metadata in order to facilitate the transfer of information.
Other methods to automate the use of XML as a communication format are also evolving. XML-RPC (Remote Procedure Calling protocol) is one developing protocol to automate the communication between systems using XML as an intermediary format (Winer 1999). Rather than importing and exporting the data the systems would use dynamic requests of each other using a hub format and the appropriate communication protocol information. This would allow universal communication between heterogeneous systems. The protocol is designed to go far beyond database communication and will in theory allow distributed applications to integrate together into a complete application. Other technologies that are evolving to provide the same kind of interoperability and interactivity include SOAP (Simple Object Access Protocol). This uses XML as an intermediary format between systems as well (Shribner 2000). Products such as .NET are beginning to be developed to take advantage of these communication protocols.

**Browser viewing**

By placing database information into XML you can show the information on the World Wide Web through a browser that supports XML. Currently not all browsers are completely XML-friendly but there is definitely a push to ensure that they do. The way the metadata is presented within the browser can be changed to only show the information that is relevant to the viewer. As the need to express the information in different formats arises, new ways to display information can be created using new markup schemes. It is also possible to translate the XML into different formats for the browser using something like the XHTML or HTML to combine existing and new feature sets (Pemberton et al 2000). The XSLT specification is one method of providing the XHTML. This will allow a middleware product to interpret queries and format the XML for display in HTML-based browser.

Using XHTML also allows for the original XML data to be viewed by browsers on alternate platforms. Estimates are that by 2002 up to 75% of Internet document viewing will be occurring on these alternate platforms (Pemberton et al 2000). As new browsers are designed for PDAs, cell phones and other mobile devices the data will be able to transform itself to meet the requirements of these new browsers.

**Document-based systems**

The XML could be parsed out to form documents that could then be handled within a document-based system. This would mostly be relevant for text-based documents that were described within the educational object repository. XML provides the foundation for managing not only documents but also the information components of which the documents are composed. By describing the structure of the document, not just the object that contains the document, the document can be handled in its component parts.

This approach will allow document systems to handle objects in a number of useful ways. There will be a certain degree of reuse or multi-purposing possible by sharing the individual components of a document without storing the information in multiple places.
(Freter 2000). In a similar way useful information components may be harvested out of a document even when the rest of the document becomes outdated or irrelevant.

The individual components can be identified, manipulated and translated into other formats as suits the needs of the user. This will allow components to be modified to match new languages, technical formats or proprietary formats.

**XML Tools**

There are numerous XML toolkits and software available on the market currently. As XML becomes recognized as a very useful format for information more and more tools are being designed or altered to take advantage of the new format. The ability to convert database information into XML will also allow the data to take advantage of the analysis and formatting packages available.

**Conclusions**

XML is quickly becoming a popular format for many Internet applications. It is both human and machine-readable and allows communication to occur between a variety of applications and platforms. Currently there aren’t many of these tools designed for use with educational object repositories and the metadata that organizes them. Most repositories function perfectly in storing, organizing and retrieving educational objects. It may not be immediately obvious why educational metadata might want to therefore take advantage of XML when all the functionality required of the repository already exists.

There are two areas that will drive the need; one is the client using the repositories and the second is other repositories themselves. Each client or user has a different focus on what they want the system to accomplish. This includes features, content and education-based goals. Currently this has lead to several different implementations. If you design the system to change easily and quickly with the needs of the user you only need one implementation of the system. XML is one solution that will allow you to change the output of the system to meet the needs of the client without having to build new functionality into the system. Other repositories are the other motivating factor in using XML in repositories. They represent a considerable amount of work both in designing their content and indexing it all with metadata. They are the fastest and cheapest way to build up the existing body of well-indexed educational content. XML is one of the best ways that institutions can begin to share their content break down many of the current barriers to communication and collaboration. It may provide a solution to many of the problems that currently limit the use and adoption of on-line educational objects.

**REFERENCES**

Freter, Todd
2001 *XML: Document and Information Management*
Shanmugasundaram, Jayavel, Kristin, Tuft; He, Gang; Zhang, Chun; DeWitt, David; Naughton, Jeffrey

Scribner, Kenn
2000  XML and SOAP--Essential to .NET  

2000  *XHTML™ 1.0: The Extensible HyperText Markup Language*  
*A Reformulation of HTML 4 in XML 1.0*  [http://www.w3.org/TR/xhtml1/](http://www.w3.org/TR/xhtml1/), accessed on July 10, 2001

W3C
2000  *Extensible Markup Language (XML) 1.0 (Second Edition)*, edited by Bray, T.; Paoli, J.; Sperberg-McQueen, C. M.; Maler, E.,  
[http://www.w3.org/TR/REC-xml](http://www.w3.org/TR/REC-xml), accessed on July 12, 2001

1999  *XSL Transformations (XSLT) Version 1.0*, edited by James Clark,  
[http://www.w3.org/TR/xslt](http://www.w3.org/TR/xslt), accessed on August 15, 2001

Winer, Dave