# Adaptive Hypermedia Educational System based on XML Technologies

Yeongtae Baek
Kimpo College, Kimpo, Korea
hannae@kimpo.ac.kr

Changjong Wang
Inha University, Inchon, Korea
cjwang@inha.ac.kr

Sehoon Lee
New Jersey Institute of Technology, Newark, USA
Seihoon_lee@hotmail.com

Abstract : In this paper we propose adaptive hypermedia educational system using XML technologies such as XML, XSL, XSLT, and XLink. Adaptive systems are capable of altering the presentation of the content of the hypermedia on the basis of a dynamic understanding of the individual user. The user profile can be collected in a user model, while the knowledge about the domain can be represented in the form of a concept based domain model. So we have defined two different markup languages using XML. And for adaptivity of system, adaptive presentation of the data comes using XSL and adaptive navigation of link comes using XLink.

## Introduction

With the rapid advances in WWW interactive technologies, the use of Internet-based distance learning tools is rapidly growing. Most of these products are nothing more than a network of static hypermedia pages. In fact the domain knowledge implicit in traditional educational hypermedia is well defined and carefully structured and provides an only learning path optimal for a generic average student [Ekl1997]. Otherwise, a Web application, which is designed with a particular class of users in mind may not suit other users. Moreover, "static" hypermedia assume that the users can make sensible decisions about when to use navigation tools, about when to proceed in the learning process, about when they need an explanation, etc. [Bru1998]. This could be a problem for those users who access the hypermedia through the Internet and that can't have a teacher at their disposal.

Adaptivity is the feature of hypertext and hypermedia that allows one to adapt the contents to the user needs[Ada2001]. Adaptive hypermedia systems modify the presentation of the domain knowledge according to the user profile. This mechanism permits to personalize the Hypermedia in terms of contents and of navigation tools for each user.

The focus of this paper is on the general architecture and the implementation issues of a adaptive educational hypermedia system. This system has been implemented by using the functionalities provided by XML(XSL,XLink) in order to stress the separation of the

information content from presentation. The basic idea is to define a general hypertext structure in order to create pages dynamically using a structured description of the domain knowledge and a model of the current user.

## Adaptive hypermedia technologies and XML techniques

Adaptive hypermedia systems build a model of the goals, preference, and knowledge of each individual user, and use this model throughout the interaction with the user, in order to adapt to the needs of that user[Bru 2001]. In adaptive hypermedia literature they are referred respectively as adaptive presentation and adaptive navigation support

XML is designed to be the data format for the web, and at the time of writing is being used by many thousands of web-based applications. XML is a bit like HTML, the format used for web documents, but it's focused on representing data, rather than describing the presentation of data. Presentation of the data comes using XSL and navigation of hyperlink comes later, using XLink. Also XML is independent from a specific platform and operating system and provides the DOM(Document Object Model) as a platform - and programming language -independent API for the development of the necessary tools. XML grammars can be constructured in a way that they easily can be extended by new or alternate elements. There already exists a metadata scheme for teaching resource – IMS projects[IMS2001].

### Adaptive presentation and XSL

Adapting the presentation of information within a page is most often performed as a manipulation of text fragments. The aim of these manipulation can be[Bra1999]

- Providing prerequisite, additional or comparative explanations. Two techniques that are used to provide such explanations are conditional inclusion of fragments and stretchtext.
- Providing explanation variants. The same information can be presented in different ways, depend on values in the user model.
- Reordering information. Depending on user model values the order in which information items are presented may have to be altered.

The structure of document can be made more visual and the full range of formatting options such as, fonts, colors, and layout can be used to create a powerful presentation of the data. Tables can be used where appropriate, and implicit relations can be made visible. Sorting and other rearrangements of the data can be used to create a page that is easy to navigate. Other data analysis could be used to create useful information about the data that would be difficult and in some cases nearly impossible to determine by looking at the XML file.

The accompanying figure shows a rendering of the data in XML file. The web page, result of rendering, is a fairly straightforward view of the data without a lot of enhancement. It is important that these web page are automatically generated from the original XML file. Thus the presentation is always up to date and is applicable to any XML file in any domain.

The conversion from XML to web page does where are two possibilities.

- Client-side conversion directly in the web browser. The browser read the XML and rules for rendering the XML as HTML.
- Server-side conversion in conjunction with the web server. When a request is made for a document, the web server can do the conversion and return HTML file, thus avoiding and need for XML-aware browsers.

There are many ways to render XML as HTML. One such method is the use of XSL(extensible Stylesheet Language)[XSL01]. XSL includes both transformations and formatting. An XSL file, which itself is an XML, is used to specify how to translate the bits and pieces of another XML file to HTML or to an XML language.

Although the most logical and likely way to present XML files is by using XSL-like techniques and HTML, we are by no means limited to these. We can use other rendition techniques(cascading style sheets) and other presentation language(such as PDF).

XSLT is a language for describing the transformations of an XML document to another XML document or a plain text file. An XSLT specification itself is an XML document whose element type names have the "xsl" namespace prefix. As with XPath, we can give only a brief overview of XSLT. XPath is a language for identifying and selecting parts of an XML document. The most powerful feature of XSLT if XPath, which provides a concise and powerful means of extracting exactly the information we want from an XML document. However XPath is intended simply to access XML information and not change it. The DOM API will allow us to modify and create new XML structure.

XSLT has a lot of element easily for transform another document. The XSL template rule describes how to transform an XML element. The form of the rule is

```
<!-- Category: top-level-element -->
<xsl:template
  match = pattern
  name = qname
  priority = number
  mode = qname>
  <!-- Content: (xsl:param*, template) -->
</xsl:template>
```

The pattern is an XPath node-set expression that defines the set of potential nodes that the template can be applied to. Each node that matches is replaced with template. There are two instructions in XSLT that support conditional processing in a template: `xsl:if` and `xsl:choose`. The `xsl:if` instruction provides simple if-then conditionality; the `xsl:choose` instruction supports selection of one choice when there are several possibilities.

```
<!-- Category: instruction -->
<xsl:if
  test = boolean-expression>
    <!-- Content: template -->
  </xsl:if>
```

Other XSL element are such as value-of, for-each, apply-templates, etc.

**Adaptive navigation support and XLink**

Adaptive navigation support techniques are almost manipulation of links that are presented within nodes is typically done in one or more of the following ways[Bra1999].

- Direct guidance. The destination of the link is the node which the adaptive hypermedia system determines to be most appropriate.
- Link annotation. Link anchors are presented differently depending on the relevance of the destination.
- Link hiding. Link leading to inappropriate or non-relevant information are hidden.
- Link disabling. Inappropriate links are disabled.
- Link removal. Inappropriate links and anchors are simply removed.
- Map adaptation. Some hypermedia systems provide a graphical presentation of the link structure. Such maps can also be subject to adaptation.

The XLink(XML Linking Language) is intended to provide linking functionality similar to that provided by the HTML/XTHML <a> element, but also provides a range of more complex linking structure. XLink provides a framework for creating both basic unidirectional links and more complex linking structures. It allows XML documents to:[XLink2001].

- Assert linking relationships among more than two resources
- Associate metadata with a link
- Express links that reside in a location separate from the linked resources

An important application of XLink is in hypermedia systems that have hyperlinks. A simple case of a hyperlink is an HTML A element, which has these characteristics:

- The hyperlink uses URIs as its locator technology.

- The hyperlink is expressed at one of its two ends.
- The hyperlink identifies the other end (although a server may have great freedom in finding or dynamically creating that destination).
- Users can initiate traversal only from the end where the hyperlink is expressed to the other end.
- The hyperlink's effect on windows, frames, go-back lists, style sheets in use, and so on is determined by user agents, not by the hyperlink itself. For example, traversal of `A` links normally replaces the current view, perhaps with a user option to open a new window.

XLink's namespace provides global attributes for use on elements that are in any arbitrary namespace. The global attributes are `type`, `href`, `role`, `arcrole`, `title`, `show`, `actuate`, `label`, `from`, and `to`. XLink has a `type` attribute from the XLink namespace whose value is one of "simple", "extended", "locator", "arc", "resource", "title", or "none".

XLink offers two kinds of links such as extended and simple link. Extended links offer full XLink functionality, such as inbound and third-party arcs, as well as links that have arbitrary numbers of participating resources. As a result, their structure can be fairly complex, including elements for pointing to remote resources, elements for containing local resources, elements for specifying arc traversal rules, and elements for specifying human-readable resource and arc titles. Simple links offer shorthand syntax for a common kind of link, an outbound link with exactly two participating resources (into which category HTML-style `A` and `IMG` links fall). Because simple links offer less functionality than extended links, they have no special internal structure.

## Design of adaptive educational hypermedia system

….