

Multimedia Presentation Components in E-commerce

Simon S.Y. Shim¹, Jerry Z. Gao, and Ying Wang

Internet Technology Laboratory²

Department of Computer, Information, and System Engineering

San Jose State University, San Jose, CA 95192, USA

ABSTRACT

Synchronized Multimedia Integration Language (SMIL) is a recommendation developed by the Synchronized Multimedia Working Group in the World Wide Web Consortium (W3C). SMIL is a simple and standard way to specify a timeline based synchronized multimedia presentation over the Internet. It is a declarative authoring language based on Extensible Markup Language (XML) to define language-specific data types and tags. Since it is based on XML, it can be included in XML based catalog and data interchange in order to provide multimedia presentations. Instead of using a proprietary multimedia document types, SMIL elements are standard based way to include them. We discuss different ways to include multimedia presentations to Web based catalog. We also describe a component based SMIL player written in Java. Different components can put together to create a multimedia player. The player can run on major web browsers as a Java applet.

Keywords: SMIL, XML, JMF, Multimedia, Synchronization, Internet

1. INTRODUCTION

Most of the web-based catalog today are created using Hypertext Markup Language (HTML). HTML uses a set of tags to arrange different text-oriented language components in a web page. Even though HTML is fairly straightforward to use, it is not flexible because of: 1) the inability to specify precise look-and-feel like PostScript or PDF, 2) the predefined set of markup tags, and 3) the rarely human-readable syntax. A web site with dynamic

contents such as multimedia presentation can be implemented only with proprietary formats and mechanisms. The end result is incompatible specifications. Though tons of web pages available today, few people are really into creating synchronized multimedia presentations. Hence it is difficult to incorporate multimedia contents in an electronic catalog.

Electronic Document Interchange (EDI) has been popular recently in ways to exchange various documents and to conduct business transactions. In order to exchange documents and catalog information, they have run the same or compatible software on participating organizations. However it is often the case that software are incompatible among the parties involved in transactions. The basic unit of these transactions is the data element of which describes a basic element. The data element should be interpreted uniformly among involved parties and know how to manipulate them. However it is a difficult task to create a uniform standard which is also flexible. A set of fixed tags often limit parties to add new functionality to meet the needs of changing business rules. The parties involved have to agree on the format and meaning of data tags to interpret documents correctly on the other side. XML (extensible Markup Language) is emerged as the leading language in EDI and e-commerce to solve the incompatibility problem. For instance, Commerce XML (cXML) [23] provides the protocol and data interchange formats for e-commerce in XML. cXML attempts to provide a platform independent solution that companies can agree upon the interpretation of data elements and data interchange protocol.

In the standard of XML/EDI, XML is used to define data elements and XSL is used to define data presentation. However this protocol does not include any multimedia support for video and audio. These multimedia elements

¹ This work was supported, in part, by the grant from NSF CISERI grant under EIA-9818335. Corresponding author's email address is sishim@email.sjsu.edu

² Internet Technology Laboratory is sponsored by Intel Corporation, Microsoft Corporation, Informix Corporation, and Wytec Inc.

often involve time-line to display presentations. This paper addresses issues to relate to multimedia data elements and multimedia presentations in XML which is becoming a dominant language for e-commerce. E-commerce applications may require to insert multimedia elements in EDI documents or on-line catalog. Our approach is to use existing multimedia synchronization standards, called SMIL. We also discuss issues related to display of multimedia presentations. A catalog contains product information. Some of the products may be presented using a multimedia presentation. Multimedia presentation on Web browser may require to install one of the many popular plug-ins for displaying. These plug-ins needs to be installed before the presentation is played. Unfortunately different players are not compatible. Some players support the multimedia presentation standard such as SMIL (Synchronized Multimedia Integration Language, It presents an open standard for flexible and fully functioning multimedia presentation with strong industry support.) [19] and some don't. Hence the multimedia presentation may be played differently or does not displayed at all. Because of these incompatibility and plug-in installation issues, we developed a component based SMIL player in Java. Java based player solves the above mentioned issues since Java applet can be downloaded on demand. It is also platform-neutral.

In the following section, we are going to present a brief overview of related standard efforts. In Section 3, we described a few related works. Section 4 concentrates on multimedia support on catalog. Section 5 discusses the component based SMIL player. Finally, we conclude this work in Section 5.

2. STANDARD OVERVIEW

2.1. XML

XML is an official recommendation by the W3C in early 1998 [18]. XML addresses the issues where many proprietary solutions in the past have been tried in enhancing HTML. In particular, these solutions deal with dynamic contents using plug-ins, such as ActiveX or Java applets. However they are often not fully supported by individual web browser. XML covers the functionality of HTML document as well as other document types. To get to XML from HTML, the fundamental differences between HTML and XML are as follows. HTML is 1) a language used to define the format of a document with a predefined set of tags, 2) an application of SGML (an ISO standard for markup language), 3) describing the display format of a document, and 4) not really human-readable. As opposed to HTML, XML is 1) a language used for creating markup languages (let you define your own tags),

2) a subset of SGML, 3) describing the structure of a document, and 4) usually human readable.

XML documents are made up of logical units called "Element". An XML parser is used to read XML documents and provide accesses to their contents and structures. Applications use an XML parser to access different elements inside an XML file. More over, XML fixes some of the problems in HTML. For instance, XML addresses the dead links, slow searches, and static pages of HTML.

There are three companion recommendations for XML. They are Document Type Definition (DTD), Extensible Style Language (XSL), and Extensible Link Language (XLL), which are used to specify the layout, style sheet, and dynamic links, respectively. An XML DTD defines the logical structure of a document. It specifies document's elements and their attributes as well as the relationships among those elements and attributes. DTD also describes data structures associated with a particular application. This will improve the information search by skipping unrelated application information. XSL is the language used to specify style sheets for XML documents. XSL enables an XML document to change its presentation without further interactions with a server. These features can update certain dynamic contents of a page on the web browser side. Thus, reducing the overhead of data transfers with a web server. XLL supports all the classic hypertext links, location-independent naming, bi-directional links, aggregate links (multiple sources), transclusion (a link target document appears to be part of the link source document), and attributes on links (link types).

2.2. SMIL

SMIL has its own Document Type Definition (DTD) in the Extensible Markup Language (XML) syntax [18]. Using DTD, SMIL defines unique tags to be used in a SMIL file. One can make use of SMIL-specific tags to specify the synchronization between audio, video, image and/or text components in a multimedia presentation. Since SMIL is a declarative language and conforms to XML, XML parser is used to generate an object tree. The object tree structure contains all the media objects and presentation timing information. Based on the tree, a SMIL player can render the display for the media objects described in the given SMIL document. Playback events are scheduled on a time line, and media objects are displayed either in parallel or in sequence. Using this tree representation, the reference implementation of our SMIL player utilizes the Java Media Framework (JMF) to display multimedia objects in a web browser [5]. The player is a client-side Java applet, which takes user input

for an external pre-authored SMIL file. Then, the player produces the synchronized multimedia presentation in the web browser using JMF and the core Java package.

SMIL provides the following features:

- It enables web authors to position (location, size, z-index ordering) visual media objects and assign temporal attributes (begin time, duration or end time) among one and another.
- It provides capabilities to create interactive (via hyperlinks) multimedia presentations similar to those on computer-based interactive CDs.
- Its media objects (addressable by a URL) can either reside locally or be distributed over the web.

2.3. Other Related Standards

There are some other standards that deal with multimedia such as HyTime, MHEG, and MPEG-4. HyTime becomes an ISO/IEC International Standard in 1997. HyTime is derived from SGML, and adds mechanisms to specify hyperlinks and schedule multimedia information in time and space [8]. It is similar to HTML in that, both of HyTime and HTML are applications of SGML.

MHEG (Multimedia and Hypermedia information coding Expert Group) is an ISO/IEC International Standard since November 1996 [3]. MHEG is used in the digital-TV broadcast environment. SMIL and MHEG both offer the functionality to describe a multimedia presentation in an object hierarchy. The major difference is in the language model. SMIL uses the structured way where temporal relationship is specified. On the other hand, MHEG uses an event-driven model for presentation dependencies. Since presentation dependencies are specified in MHEG, an event can fire any actions. MHEG document is encoded in ASN.1/DER, which is a complex encoding scheme. Since SMIL is XML-compliant and parsing tools are readily available, SMIL is easier to comprehend and use.

MPEG-4 (Motion Picture Expert Group), on the other hand, is another ISO/IEC International Standard down the road. It is expected to become an International Standard in early 1999 [9]. In the beginning, MPEG-4 was addressing the low bit-rate (lower than 28.8 Kbps) digital video/audio format left out by the earlier MPEG-1 and MPEG-2 standards. It has incorporated a variety of multimedia codecs for timeline-driven synchronized multimedia stream format. Its base file format is Apple's QuickTime. Since SMIL does not specify the mixed media format, MPEG-4 file could be an option for SMIL in terms of media object storage format where strong synchronization primitives are embedded.

3. RELATED WORK

This section provides overview of multimedia presentation works, not necessarily related to EDI/XML or e-commerce. The author does not aware of any existing works related to multimedia presentation and e-commerce. Multimedia presentation synchronization has been an active research topic for years. Multimedia presentation has two major reference models with respect to presentation flow and synchronization: linear (pre-orchestrated based on timeline or story-line), and hyper-dimensional (non-linear event-driven) [6, 13, 14]. The major issues include how to specify and how to implement synchronization primitives. Blakowski and Steinmetz [1] provide an excellent survey on multimedia synchronization requirements, and specifications. Many of the intended use of multimedia presentation synchronization are geared towards visual and audio aid in information dissemination and education delivery. Robinson, *et al* [11], describes an implementation of Computer-Assisted Learning (CAL) using Java and Dynamic HTML (DHTML). It intends to use WWW as a delivery medium and integrated learning environment. Yu [15, 16] proposes the Hypermedia Presentation and Authoring System (HPAS) which implements Media Relation Graph (MRG) as a Java applet. HPAS is a timeline based presentation synchronization model. The author discusses in details on a synchronization model using the MRG. It runs only on the Netscape web browser.

In order to assist W3C members in their evaluation of SMIL, CWI provides copies of their SMIL player V1.0 to work group members for the duration of specification review process. Unfortunately, the CWI SMIL player was not available to us. Bulterman, *et al* [2], discuss an implementation of a SMIL player. They develop a player and an authoring tool for SMIL. However it is written as a stand-alone application. As a consequence, it does not work with a web browser. Different versions of the player need to be created for different platforms. Real Networks [10] implemented a SMIL player using its RealPlayer G2. It is a commercial player that is capable of interpreting part of the SMIL standard. It supports a variety of proprietary streaming data types, and provides client-server synchronization. Rousseau and Duda [12] propose ways to support synchronized multimedia as an extension to HTML. They addressed the issues in tighter temporal synchronization with common time bases in order to support lip-sync.

Since the structural rules of XML are more strict than those of HTML, every XML document must be *well-formed*. A well-formed XML document is simply one that follows all of the notational and structural rules for XML. Programs that intend to process XML should reject any

input XML that does not follow the rules for being well-formed. A generic XML parser is a program or class that can read any well-formed XML at its input. They come in two flavors:

In addition to checking for well-formedness, the parser verifies that the document conforms to a specific DTD (either internal or external to the XML file being parsed).

The functionality of the XML parser is simple. The programs call the parser using API. The parser processes error detection and provides the document structure as an object tree when the given document is well-formed. Many vendors offer XML parsers in Java for free. For example, Microsoft's XML parser in Java [4] is included in Internet Explorer (IE) version 4 and later. It is a validating parser, which exposes the well-formed XML document through a simple set of Java methods. These methods support reading and/or writing XML structures. Therefore, XML provides a portable data format that nicely complements Java's portable code.

The Java Media Framework (JMF) provides functions to synchronize and display time-based data, such as audio and video, within Java applications or applets. The JMF APIs are being developed by Sun Microsystems, Inc., Silicon Graphics Inc., and Intel Corporation. It is being released in three stages: Player, Capture, and Conference APIs. Java Media Player is the first release of the JMF technologies and supports synchronization, control, and presentation of compressed streaming and stored time-based media across all Java enabled platforms. Table 1 shows the list of available JMF implementations. This is by no means complete due to growing number of platform supported. We used Intel's Java Media Player for our implementation [3].

A Java Media Player is an object that reads and displays data from a data source. It supports MIDI, WAV and AU for audio, and AVI, MPEG, and QuickTime for video. It specifies a data source, which includes locations of media objects, and the protocols to deliver media objects. JMF supports two different types of media sources: *Pull Data-Source* (the client initiates the data transfer and controls the flow of data) and *Push Data-Source* (the server initiates the data transfer and controls the flow of data). A Player's *media time* represents the current position in the media stream. Each Player has *TimeBase* that keeps track of time for itself. When a Player is started, its *media time* is mapped to its *time-base* time. To be synchronized, Players must use the same *TimeBase*.

JMF has a hierarchical model of events for programs to respond to media-driven error conditions, such as out-of-data or resource unavailable conditions. The event system also provides an essential notification protocol. When a program calls an asynchronous method on a Player, it can

only be sure that the operation is complete by receiving an appropriate event. A Java Media Player can be in one of the following six states: *Unrealized*, *Realizing*, *Realized*, *Prefetching*, *Prefetched*, and *Started*. A Player posts *Transition Events* as it moves from one state to another. The *Controller Listener* interface is derived to determine what state a Player is in and to respond accordingly.

4. MULTIMEDIA SUPPORT IN CATALOG

SMIL becomes an official recommendation by the W3C in mid 1998 [19]. In supporting multimedia presentation for E-commerce applications, there are some advantages of using SMIL compared to other standards such as MHEG and MPEG-4. SMIL documents are essentially XML documents as most of E-commerce documents are also based on XML. Another advantage is that it is simple to create a SMIL file (plain text) and to be embedded easily in any XML documents. In addition, SMIL is widely supported by industry. Some media players such as the Real Player already support it. Timing and Media objects as specified in SMIL-Boston are supported in the IE5.5 preview release. In our opinion, SMIL will be the multimedia presentation standard to be integrated into E-commerce frameworks.

A SMIL element contains a head element and a body element. The head element contains information that is not related to the temporal behavior. The body element contains information that is related to the temporal and linking behavior of the document. In the head element, layout element defines regions, where elements contained in the body element are placed. Presentation information is described in body element. A media object element in the body element describes a media object, which can be audio, video, image, text, text stream, animation, or reference. Media object elements may be arranged using parallel or sequential elements to prescribe presentation sequence in time. Two media objects are played simultaneously or one after another, which result in parallel or sequential presentations, respectively. Startup delay can be specified either in time or by events before a media object can be displayed. Hence, the end of a media object or a timer expiration event may trigger the display of other media object(s). The following is a simple example of a SMIL file.

```
<smil>
<head>
  <layout type="smil-basic">
    <root-layout height="400" width="500"
      background-color="192,192,192"/>
    <region id="left-video" left="20" top="0"
      width="180" height="150" z-index="1"/>
  </layout>
</head>
```

```

<region id="right-video" left="280" top="30"
width="150" height="150" z-index="2"/>
<region id="left-image" left="100" top="200"
width="180" height="180" z-index="1"/>
<region id="right-text" left="300" top="300"
width="180" height="30" />
</layout>
</head>
<body>
<par>
<video src="item.mov" region="left-video"/>
<seq>
<video src="newitem.mpg" region="right-video"/>

<text src="text.txt" region="right-text" dur="8s"/>
</seq>
<audio src="product.wav"/>
</par>
</body>
</smil>

```

The above example shows a SMIL file with two videos, one image, one text, and one audio object. A SMIL file includes its own default tags which are specified in the "smil-basic" file. The "smil-basic" file is a DTD file that includes the syntax and definition of default tags. Hence a XML parser will check the syntax of a SMIL file using this DTD file. The SMIL/XML file is a plain text file similar to HTML. However, the tags, such as *img*, *video*, and *audio*, cannot be interpreted in HTML. In the head element, each region is defined by specifying the coordinate position in pixel units or percentage values from its upper left corner of the viewing canvas. Each region has its own identifier that is used in the body element. The *z-index* is used to specify the depth of overlaying regions. In the body element of the example, the video "item.mov" and audio "product.wav" are played in parallel (enclosed by *par* tag) and "newitem.mpg", "image.jpg", and "text.txt" are displayed in sequence (enclosed by *seq* tag).

The simplest way to include the SMIL presentation is to use the existing smil tag in order to include the SMIL presentation to EDI or a product catalog. A catalog plays a pivotal role in e-commerce. Buyers search catalogs to retrieve product information. It is the documents to be shared among the product producers and sellers. Hence catalogs should be interoperable. Otherwise, the cost will be very high to maintain incompatible catalogs. A catalog needs to provide searchable indexes so that multiple catalogs can be searched at once for comparison shopping. Hence all suppliers' catalogs need to follow a certain standard for easy of search. A catalog often contains the details of a product which include name, description, price, etc. It often includes images,

animations or videos on a product. In order to include multimedia information, one can follow the SMIL standard or define proprietary formats. Obviously, there are advantages on using the industry standard. In order to include multimedia presentation in a catalog or an EDI document, one has to be consider interoperability and flexibility. We tried to merge different standards in e-commerce and multimedia presentations. As an example, in cXML, catalog information is encoded to be used in buyer's side. It is consisted of supplier, index and contract. The index element contains the information on product items such as itemID, and itemDetail. Inside the itemDetail, the smil tag can be inserted indicating a multimedia presentation is available on the product.

```

<Index>
<IndexItem>
<IndexItemAdd>
  <IndexItemDetail>
    <ItemID> <partID> part101 </partID>
    <ItemDetail>
      <smil>
        <head>
          <layout type="smil-basic">
            <region id="left-video"
              left="20" top="0"
              width="180" height="150"
              z-index="1"/>
          </layout>
        </head>
        <body>
          <par>
            <video
              src="www.foo.com/part101/item.mov"
              region="left-video"/>
            
          </par>
        </body>
      </smil>
    </ItemDetail>
  </ItemID>
</IndexItemDetail>
</IndexItemAdd>
</IndexItem>
</Index>

```

The above example shows an index item which contains an audio which describes an item, part101. The audio description is stored in a url and should be played along with a product image stored in the similar url. In this way, the multimedia presentation is included in the catalog protocol. A user can retrieve product information along with multimedia presentation. And the multimedia presentation will be played automatically inside a web

browser in any operating system platform. The two standards can be interoperable here because both are based on XML. On the buyer side, a SMIL player needs to be installed or downloaded on demand. The similar scenario can occur when multimedia documents are exchanged.

SMIL standard support both URL based media files and local files. Some standards such as BizTalk[24] specifies a way to support media files as a part of documents. In BizTalk, a binary file can be encoded as texts so that it can be transferred as a part of the document. The text-encoded files may require special tags and decoding program to be able to recognize special files. However this tends to increase the size of documents significantly. If media files are used only in product presentations, it will be better to include them as URLs. Hence on-demand download from the Internet may be preferred without increasing the transfer size dramatically.

The other alternative for multimedia presentations is to use timing tags as proposed in HTML+TIME [22]. HTML+TIME is the proposal to World Wide Web Consortium by Microsoft, Macromedia, Compaq/Digital and Digital Renaissance. This proposal provides the browser support of timing tags in HTML by extending SMIL concepts. Hence a browser is capable of processing time elements in Document Object Model (DOM). The time elements will enable or disable style tags and visibility of attributes. In addition, it provides <par> and <seq> tags which behaves similarly as in SMIL. It lends the support to plug-in players to coordinate different players, thus acting as a coordinating container. In this approach, the browser has to support these timing tags to provide multimedia presentations and it will be up to the browser which multimedia elements are supported. Hence this approach may only work for some browsers in a platform.

In order to integrate other languages such as HTML and XML, SMIL-Boston is proposed in November 1999 [20]. SMIL-Boston is designed to extend SMIL so that it is easier to integrate with HTML or XML. The syntax has changed to closely follow the DOM syntax. SMIL-Boston includes eight modules. They are Content Control Module, Layout Module, Linking Module, Media Object Module, Metainformation Module, Structure Module, Timing and Synchronization Module, Transition Effects Module. Some of features in HTML+TIME such as interactive timing and control of synchronization are appeared also in SMIL-Boston. In the Timing and Synchronization module, a media presentation can be scheduled and change contents of the presentation over time. The timing components can be included in any XML languages. For instance, a product catalog is scheduled to appear in the scheduled time with other

media objects. The scheduled actions are enabled by using the TimeAction tag which fires multiple actions specified in the attribute list. Timing can be applied to each individual page with differing timing structures and unique id. Hence a page can be animated based on a time line as styles or appearances changes. It also provides repeat attributes to control the repeating behavior of a presentation. Hence the same animation of documents can be replayed many times. Microsoft's IE browser 5.5 supports subset of modules of SMIL-Boston. The supported modules include Animation, Timing and Synchronization and Media Object. The IE 5.5 preview release can be downloaded from <http://msdn.microsoft.com/downloads/webtechnology/ie/iepreview.asp>. It is not clear which modules or how much of SMIL-Boston will be supported by either Microsoft or Netscape. The important issue to address is how the browser will coordinate time lines with other plug-in players such as Real player, Microsoft media player, or many other media players.

5. COMPONENT BASED SMIL PLAYER

Our approach of presenting multimedia presentation is to use a Java container. This Java container applet is central point of time based event coordination. All other players of HTML, text, image, and video have to be registered with Java container and events are fired based on the specified time line. The advantage of using Java applet is the platform and browser independence. Our player will work as long as the browser support JVM and there is no need to pre-install any plug-ins. Our early version of the SMIL player[17] has limitations on extending the player because of non-modularized design. In this section, we describe our component based SMIL player. Our design goals are to create modularized components for multimedia presentation player so that it may be extended to support SMIL-Boston. Java is chosen to provide platform independence. In order to create components for different media players, we used JavaBeans. The core of JavaBeans is a set of standard communications methods that allow a container to know what to do with a Bean without having any specific knowledge of what that Bean is and what it is supposed to do. Besides the publishing and discovery capability, JavaBeans also supports services such as event handling, persistence, and secure distributed computing. We created HTML, text, image, and video player Beans. The multimedia presentation based on the SMIL standard needs to be synchronized with different media in a time line. In order to create this time based coordination, we created a container that consists of many player components. Different players are coordinated using events inside JMF.

Different medium can be included as a media element in a multimedia presentation. One of the player's function is to support HTML document rendering and displaying. There are three HTML Renderer Components in the market. They are HotJava HTML Component from Sun, and ICE Browser 4 from ICEsoft [22]. We chose to use ICE Browser for its functionality and compact size. It can parse and render HTML documents loaded from the local file system or over HTTP connections. ImagePlayer class is responsible for rendering images and text for Preview Bean. Image rendering is simple: get image size and calculate its displaying position in "image" panel card. However, text rendering is more complicated, because sentences in text file may be longer than the "image" panel width. Video player is constructed using JMF. Currently, it supports object layout, and both sequential and parallel synchronization primitives. However, it does not support the anchor feature.

6. CONCLUSION

Multimedia technology covers a wide range of areas, including the encoding and decoding of media formats, the protocols for transmitting the media over the network, the quality of service for media objects and more. Documents will increasingly contain multimedia presentations. This may include animation, audio, and video. With the wide deployment of high speed network technologies such as xDSL and cable modem, multimedia will be prevalent in Web documents. Certainly, there are many alternatives in including multimedia objects and their synchronized presentations. We concentrate on current state-of-art in multimedia presentation for Web documents and shows examples in a on-line catalog. As the part of the continuing effort in bringing synchronized multimedia presentation to the web, SMIL provides a common ground towards open standard. Our approach uses the industry standard and creates a flexible presentation container based on the existing technologies. We tried to use the existing solutions to minimize the development time and to take advantage of future enhancements of each solution. We believe that interoperability and cross-platform support from various SMIL implementations pose a challenge in the near future.

7. REFERENCES

1. G. Blakowski, and R. Steinmetz, "A Media Synchronization Survey: Reference Model, Specification, and Case Studies," In *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 1, pp. 5-35, January 1996.
2. D. Bulterman, L. Hardman, J. Jansen, K. Mullender, and L. Rutledge, "GriNS: A Graphical Interface for Creating and Playing SMIL Documents," In *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 519-529, April 1998.
3. W. Effelsberg, and T. Meyer-Boudnik, "MHEG Explained," In *IEEE Multimedia*, vol. 2, no. 1, pp. 26-38, Spring 1995.
4. Cay S. Horstmann, Gary Cornell, "Core Java 1.1 Volume II – Advanced Features"
5. Javasoft, *Java Media Framework API*. Available at <http://www.javasoft.com/products/java-media/jmf/index.html>
6. W. Ma, Y. Lee, D.H.C. Du, and M.P. McCahill, "Video-Based Hypermedia for Education-on-Demand," In *IEEE Multimedia*, vol. 5, no. 1, pp. 72-83, January-March 1998.
7. Microsoft Corp., *Microsoft XML Parser*, <http://www.microsoft.com/xml/>
8. S.R. Newcomb, N.A. Kipp, and V.T. newcomb, "The HyTime: Hypermedia/Time-based Document Structuring Language," In *Communications of the ACM*, vol. 34, no. 11, pp. 67-83, November 1991.
9. A. Puri, and A. Eleftheriadis, "MPEG-4: An Object-based Multimedia Coding Standard Supporting Mobile Applications," In *ACM/Baltzer Journal of Mobile Networks and Applications*, vol. 3, no. 1, pp. 5-32, June 1998.
10. Real Networks, *RealNetworks G2 Player*, <http://www.real.com/>
11. D. Robinson, C. Lester, and N. Hamilton, "Delivering Computer Assisted Learning across the WWW," In *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 301-307, April 1998.
12. F. Rousseau and A. Duda, "Synchronized Multimedia for the WWW," In *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 417-429, April 1998.
13. J.A. Schnepf, J.A. Konstan, and D.H.C. Du, "Doing FLIPS: FLeXible Interactive Presentation Synchronization," In *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 1, pp. 114-125, January 1996.
14. J.A. Schnepf, Y. Lee, D.H.C. Du, L. Lai, and L. Kang, "Building a Framework for Flexible Interactive Presentation," In *Proceedings of the 1996 Pacific Workshop on Distributed Multimedia Systems*, pp. 296-305, Hong Kong, June 1996.
15. Jin Yu, "A Simple, Intuitive Hypermedia Synchronization Model and its Realization in the Browser/Java Environment," *Technical Note 1997-027a*, Digital Equipment Corporation Systems Research Center, Palo Alto, CA, April 1998. Available at <http://gatekeeper.dec.com/pub/DEC/SRC/technical-notes/abstracts/src-tn-1997-027.html>

16. Jin Yu and Yuanyuan Xiang, "Hypermedia Presentation and Authoring System," In *Proceedings of the 6th International World Wide Web Conference*, pp. 153-164, April 1997. Available at <http://gatekeeper.dec.com/pub/DEC/SRC/technical-notes/abstracts/src-tn-1997-003.html>
17. J. Xia, S. Shim, Ying Wang, and Y. Lee, "Implementation of A Synchronized Multimedia Integration Language Player," *SPIE Internet Imaging '99*, January 26 - 29, 1999, Vol. 3648, pp. 382 - 389.
18. World Wide Web Consortium, *Extensible Markup Language (XML) 1.0 Specification*, W3C Recommendation REC-xml-19980210, February 10 1998. Available at <http://www.w3.org/TR/REC-xml>
19. World Wide Web Consortium, *Synchronized Multimedia Integration Language (SMIL) 1.0 Specification*, W3C Recommendation REC-smil-19980615, June 15 1998. Available at <http://www.w3.org/TR/REC-smil/>
20. World Wide Web Consortium, SMIL-boston November 1999. Available at <http://www.w3.org/TR/smil-boston/>
21. World Wide Web Consortium, Timed Interactive Multimedia Extensions for HTML(HTML+TIME). Available at <http://www.w3.org/TR/NOTE-HTMLplusTIME>
22. ICE Browser 4 from ICEsoft, <http://www.icesoft.no>
23. cXML version 1.0 specification, <http://www.cxml.org>
24. BizTalk framework specification 1.0a, January 2000, <http://www.biztalk.org>