A Live TV-Quality Distant learning Multimedia Presentation System for Education

Sheng-Tun Li

Department of Information Management
National Kaohsiung First University of Science and Technology
1 University Road, Yenchao, Kaohsiung
Taiwan 824, R.O.C.
stli@ccms.nkfust.edu.tw

Shu-Ching Chen

School of Computer Science
Florida International University
Miami, FL 33199
chens@cs.fiu.edu

Mei-Ling Shyu

Department of Electrical and Computer Engineering
University of Miami
Coral Gables, FL 33124
shyu@miami.edu

Abstract

In this paper, an abstract semantic model called multimedia augmented transition network (MATN) *to model a live TV-like multimedia presentation system for distance learning education purpose is presented. Unlike the original design for CISCO IP/TV system that is built upon Microsoft ActiveMovie technology, a Java-based viewer called* JTViewer *is developed to provide the cross-platform capability for CISCO IP/TV system. In the* JTViewer *system, different media streams need to be synchronized and displayed in real-time. In order to meet this requirement, The MATN model is the underlying semantic model for the presentation in the proposed framework since it provides the visualization of control structure and a good programming data structure for the implementation. An example of distance learning via multimedia presentations is given to illustrate how the* MATN *model serves as the underlying semantic model for the* JTViewer *system.*

1. Introduction

The recent progress in multimedia technologies has made multimedia data more and more prevalent. Because of the variety of media data need to by transmitted and synchronous for the presentation, a multimedia system is more complicated than the traditional system development. This has created a need to develop abstract semantic models to represent and model the different requirements for media data such as video, audio, text, and image data for multimedia systems. There are two main requirements for an abstract semantic model. First, it should be able to effectively provide visualization of control structure for multimedia systems. Second, it should be a good programming data structure for implementation to control multimedia systems.

Multimedia information has been used in several applications including education, computer-based training (CBT), manufacturing, medicine, entertainment, video conference, etc. In particular, distance learning has become the mainstream of computer-based training and education. For example, there are 247 articles with the distance learning subject published on IEEE/IEE Library from 1997 to 1999 [8]. Because of the seamless integration of the World Wide Web (WWW) and the emerging Java technology, the universal accessibility to diverse distance learning services becomes feasible [1, 7, 15, 17, 20]. The WWW provides a cross-platform consistent visual user interface for accessing information, and the Java technology allows the application code (applets) to be downloaded over the Internet and run on any Java-compliant Web browsers.

For distance learning, the delivery of TV-like video is very important. In general, there are two directions toward the convergence of PCs and TV: empowering PC with the functionality of TV and vice versa [9]. The first direction focuses on developing the architecture-neutral delivery of TV-quality programs. The other direction is driven by the need of set-box top (STB) and the upcoming WebTV. In our previous studies, a Java-based integrated distance learning paradigm was proposed [13] and a Java JMF-based viewer called *JTViewer* for delivering real-time TV-quality courseware was developed [14]. The *JTViewer* system aims at empowering PC with TV's functionality and improving the delivery of TV-quality courseware by appealing to Cisco's IP/TV solution. Cisco IP/TV mainly deploys Microsoft ActiveMovie and Direct Media technologies, and can be run only on Windows 95/98/NT platforms. In order to resolve this cross-platform issue, the *JTViewer* system is implemented using Java technology so that it can be straightforward ported to any Java-compliant environment to provide another feasible solution for Cisco IP/TV system. Hence, it is a Java-centric TV-quality distance learning system. As mentioned in [12], most of the streaming-video technologies require users to download plug-in or stand-alone client programs that need to be compatible with the client platform. By using the Java applets, this problem can be solved by downloading the content via a Java applet in a Java-enabled browser. The *JTViewer* system that uses the Java applet approach and has been implemented at National Kaohsiung First University of Science and Technology (NKFUST) is developed for this purpose. The system aims at digitizing and distributing video courseware tapes recorded in a synchronous distance learning classroom to improve curriculums, to provide another channel for learning, and to complement synchronous/asynchronous learning.

In the *JTViewer* system, different media streams need to be synchronized and displayed in real-time. Therefore, the development of a good abstract semantic model for such a distance learning system is very crucial. Toward this end, the *multimedia augmented transition network (MATN)* that serves as the underlying semantic model for the multimedia distance learning presentation system in *JTViewer* is presented in this paper. An MATN consists of a finite set of state nodes connected by the labeled directed arcs. The arcs in an MATN represent the time flow from one state to another, which is a good structure for the multimedia presentations. The MATN model has been used to model multimedia presentations [2, 3], multimedia database searching, the temporal, spatial, or spatio-temporal relations of various media streams and semantic objects [3], and multimedia browsing [4].

The MATN model meets the two requirements of an abstract semantic model for multimedia information systems since it provides a simple visualization control structure and a good programming data structure for implementation of the multimedia distance learning presentation system in the proposed *JTViewer*. The inputs for the MATNs are modeled by multimedia input strings. Multimedia input strings provide an efficient means for iconic indexing of the time durations and time instants in multimedia presentations.

The remaining of the paper is as follows. Section 2 briefly gives the high-level system architecture for the proposed distance learning multimedia presentation system. The technologies integrated in the proposed system and the benefits of the proposed system to distant learning for education are also included in Section 2. The proposed *JTViewer* system along with the MATN model are introduced in Section 3. In Section 4, a distance learning multimedia presentation example is presented to illustrate how the MATN model is used as the underlying semantic model for the presentations in the *JTViewer* system. Section 5 concludes this paper.

2. The Distance Learning Multimedia Presentation System

2.1. The High-Level System Architecture

Figure 1 gives the high-level system architecture for the proposed distance learning system, where a set of the proposed *JTViewers* is located in both Group A and Group B. The goal of this distance learning multimedia presentation system is to empower PC with TV's functionality. In order to deliver TV-quality distant learning services, several technologies such as Java, Java Media Framework (JMF), IP multicast, RTP, and media streaming need to be integrated [6]. The integration of these technologies makes it possible for the proposed system to deliver and present TV-quality programs efficiently on high-speed IP-backbone. In addition, the *JTViewer* is a novel architecture based on Java JMF technology [10] for playing live TV-like video, and allows the audience to view RTP video/audio live programs through IP multicast transmissions [21].

As can be seen from Figure 1, the RTP media contents come from the optical driver, camera, or cable TV. The IP/TV Server delivers the contents to a large audience using multicast transmission via multicast-enabled routers. A client host in Group B is assigned to run as the Relay Agent and to communicate with the Tunnel Agent in Group A through tunnel. The Relay Agent autonomously retrieves unicast RTP streams from the Tunnel Agent and re-delivers the streams using multicast transmission for those viewers separated by the non-multicast routers. The approach of Relay Agent, playing as multicast routing software in Multicast Backbone (MBone), is more cost-effective than Cisco's since a dedicated IP/TV Server is not required to be established.

Currently, there are more than eighty streaming media products on the market [18], for example, RealSystem G2 from RealNetworks, NetShow Services and Theater Server from Microsoft and IP/TV from Cisco. Recently, RealPlayer for Java that is released by RealNetworks and a wrapper around the RealSystem G2 provides a hybrid solution allowing the ability to play G2 objects via the JMF API [19]. Microsoft solution follows the standard MPEG-4 for video compression/de-compression. Then, Cisco's IP/TV that is a distributed application delivering TV-quality streaming video and audio on a high speed IP-based LAN or WAN supports a number of stream types including MPEG-1, H.261 and Indeo [5].

2.2. The Kernel Technologies

The underlying kernel technologies that are employed

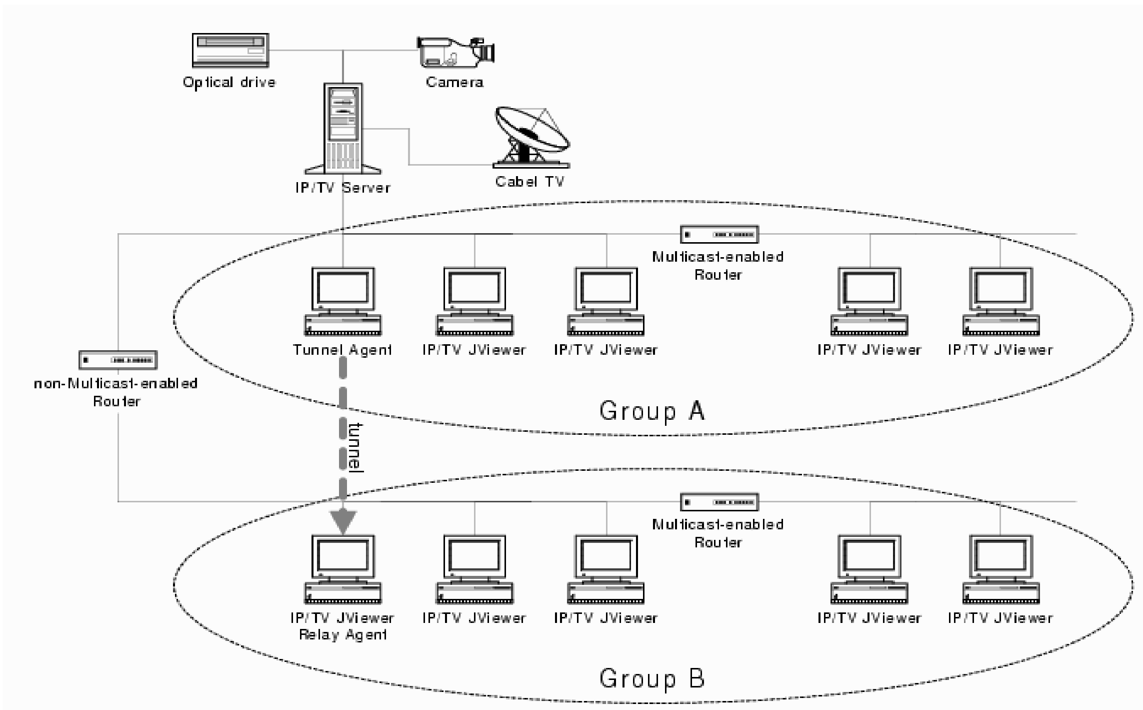

Figure 1. The high-level system architecture for the proposed distance learning multimedia presentation system, where the *JTViewers* are located.

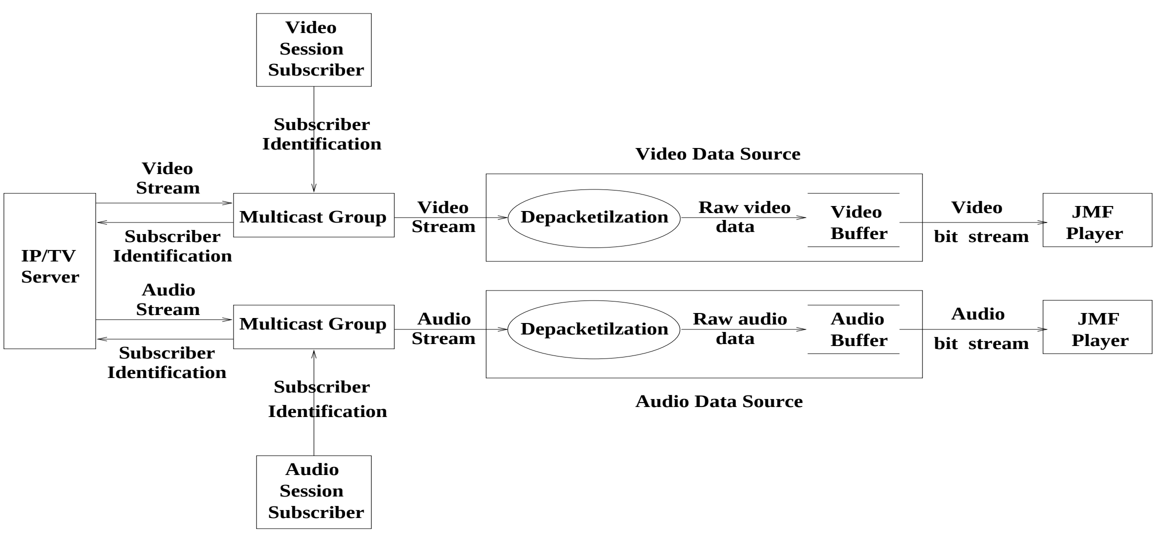

Figure 2. The *JTViewer* system architecture.

in developing the proposed multimedia presentation system include IP multicast, RTP, Java programming language, Java Media Framework (JMF), and media streaming.

IP multicast is one of the two common ways to transmit data on the Internet. IP multicast adopts a point-to-multipoint way in which one host sends a message to a group of destination hosts. Another approach is IP unicast that involves a point-to-point communication [6]. In multicasting, the hosts in a subnetwork join as the members of a specified multicast group for receiving the data packets delivered by a source host (i.e., the service provider). Only a single copy of the packets needs to be transmitted for all members in the group so that the computation load of the source host can be alleviated and the network bandwidth can be reduced. Hence, multicasting is of great help for multimedia data such as audio or video.

RTP is a transport protocol for delivering real-time multimedia streams over multicast or unicast IP networks [21]. RTP is usually carried by UDP in order to efficiently transmit data under time constraints so neither the resource reservation nor quality of service (QoS) is guaranteed. RTC is accompanied with RTPC (Real-time Control Protocol), which periodically sends information about the quality of the packet transmission to all participants in the RTP session.

Initially, the Java programming language was designed for small consumer devices. When it is coupled with the WWW for the development of distribution applications such as distant learning, it becomes more and more prevalent [1, 7, 15, 17, 20]. The Java programming language has several distinct features such as architecture neutral, interoperability, user interaction, concurrency, etc. The Java technology has been attracted a considerable attention in multimedia computing since platform independence can be achieved.

A unified architecture, messaging protocol and programming interface for playback, capture and conferencing of compressed streaming and stored timed-media including audio, video, MIDI, and MPEG-1 across all Java enabled platforms is supported by the set of classes in the Java Media Framework (JMF) API. The JMF API supports most standard media content types such as AU, WAV, MIDI, MPEG-1, QuickTime, and AVI. There are three packages in the JMF API that are *Player, Capture* and *Conference* for manipulating time-based media [10]. Two versions of JMF are currently available, which are *JMF1.1FCS* and *JMF2.0ea*. *JMF1.1FCS* supports RTP for playing broadcast and multicast media for limited media types including DVI, GSM, G723, and G711 except MPEG-1. Although *JMF2.0ea* improves MPEG-1 in which Layer 1&2&3 (MP3) is also supported, MPEG-1 video is still not provided. With Java Media Player, media players can be designed to receive and present from sources such as a local file (via FILE), a network file (via HTTP), or streaming sources (e.g., broadcast media, multicast media, or VOD).

Media streaming technology allows the users on the WWW to view or hear the media without having to wait until the whole media is downloaded. In general, there are three components in a Web-based stream system – the encoder, the streaming server, and the player. The encoder encodes the source media into one specific stream format, the streaming server delivers the streaming media upon client's request, and the player decodes and plays media streams on the client side. Some existing streaming media products on the market are RealSystem G2 from RealNetworks, NetShow Services and Theater Server from Microsoft and IP/TV from Cisco. Cisco's IP/TV is a distributed application delivering TV-quality streaming video and audio on a high speed IP-based LAN or WAN. IP/TV consists of three components – the IP/TV Content Manager, IP/TV Server, and IP/TV Viewer. The IP/TV Content Manager allows administrators to manage scheduled and on-demand programs and to configure or perform load balancing IP/TV Servers from standard Web browsers. The IP/TV Server handles multicasting scheduled programs and unicasting on-demand programs. Then the IP/TV Viewer allows the clients to watch video/audio streams chosen from a list of programs available on the IP/TV Server.

2.3. The Benefits to the Distant Learning for Education

The proposed TV-quality distant learning multimedia education system can be beneficial to the schools. The benefits can be summarized as follows.

- *Platform independence.* Though Cisco IP/TV provides a solution to the delivery of TV-quality courseware, it mainly uses Microsoft technologies and can be run only on Windows 95/98/NT platforms. However, each school may have its existing computer infrastructure. Therefore, the Cisco IP/TV solution may result in the situation that the school cannot reuse its current computer setting belonging to other vendors. On the other hand, the proposed *JTViewer* system can solve the cross-platform issue by using the Java technology.

- *Synchronous and asynchronous learning.* The proposed system has the capability to digitize the recorded video courseware tapes in a synchronous distance learning classroom. Students can retrieve the digitizing courseware to study based on their times and needs. In addition, the proposed system can relay on-line TV programs to the classroom directly.

- *High-quality learning.* The *JTViewer* system can display the high-quality multimedia presentations on the computer screens. Students can watch the programs smoothly on the screens so that they are willing to use this system to learn new topics. Therefore, it will stimulate the students' learning motivations to achieve the education purpose. Based on our observation, the number of students using the proposed system increases a lot after this system was installed at NKFUST. Especially, we found that many students use this system continuously instead of using once and giving up.

- *Reasonable cost.* By using our proposed design, the schools can afford to building up the proposed distant learning system under a reasonable cost. That is, we provide a viable solution for the schools to have a high-quality distant learning environment. In addition, the platform independence property will greatly reduce the maintenance cost since the schools do not need to maintain different softwares for different platforms.

3. The IP/TV JTViewer

In this section, the proposed *JTViewer*, a live TV-like multimedia presentation system for distance learning education purpose, is introduced. In general, distance learning services can be delivered in three ways: synchronous (real-time), asynchronous (on-demand) and hybrid of both. Synchronous distance learning systems provide live lecture contents as in the traditional classroom. Asynchronous systems offer archived lectures by using Web and/or streaming technologies. Hybrid systems supply complementary the above services. The *JTViewer* system provides a Java-centric TV-quality viewer to complement synchronous/asynchronous distance learning.

3.1. The JTViewer System Architecture

Figure 2 depicts the *JTViewer* system architecture using the object modeling technique (OMT) methodology [16]. The *JTViewer* allows audience to view RTP video/audio live programs through IP multicast transmissions. As shown in Figure 2, the actor Cisco IP/TV Server runs on Windows NT and contiguously transmits demultiplexing video/audio streams to multicast groups. Periodically, subscriber identification including name, e-mail and IP is sent to the multicast groups by the Video/Audio Session Subscriber. The process of Depacketization and Video/Audio Buffer constitutes Video/Audio Data Source which inherits from the class PullDataSource defined in JMF so JMF Player can treat video/audio raw data as coming from reliable data sources such as FILE or HTTP. Finally, the depacketized raw data in the buffer must be converted into bit streams for playing using JMF Player.

The proposed architecture of the *JTViewer* has been experimented on NKFUST campus network environment whose backbone is 155 Mbps ATM. Cisco IP/TV V2.0 is installed on a Windows NT in Computer Center. NKFUST signs a contract with a Cable TV provider for multicasting TVBS-N news 24 hours daily. Figure 3 gives a snapshot from the proposed *JTViewer* for live video on Sun Ultra 60. The screen of the *JTViewer* on the other building (as Group B in Figure 1) in which RTP multicast streams are delivered via the Relay Agent is shown in Figure 4. On the average, the *JTViewer* consumes 1Mbps bandwidth with an acceptable display rate of 21 frames per second.

Figure 3. A snapshot from the *JTViewer* for live video on SUN SPARC workstations.

3.2. The MATN Model

The *augmented transition network (ATN)*, developed by Woods [22], has been used in natural language understanding systems and question answering systems for both text and speech. The MATN model is based on the ATN with the major modifications to be used as the underlying semantic model to model the multimedia distance learning presentation system in *JTViewer*.

The MATNs are left to right models that are used to model multimedia presentation sequences. An MATN can be represented diagrammatically by a labeled directed graph and it consists of a finite set of state nodes connected by the labeled directed arcs. The arcs in an MATN represent the time flow from one state to another, which is a good structure for the multimedia presentations since a multimedia presentation consists of media streams displaying together or separately across time. An arc represents an allowable transition from the state node at its tail to the state node at its head, and the labeled arc represents the transition func-

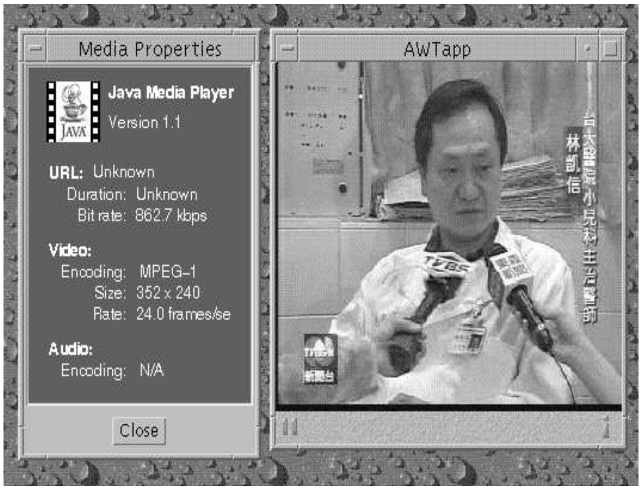

Figure 4. The video outlook sent by Relay Agent.

tion. An input string is accepted by the grammar if there is a path of transitions which corresponds to the sequence of symbols in the string and which leads from a specified initial state to one of a set of specified final states.

In MATNs, states are represented by circles with the state name inside. The state name is used to indicate the presentation or subnetwork being displayed (to the left of the slash) and which media streams have just been displayed. The state name in each state can tell us all the events that have been accomplished so far. Based on the state name, we can know how much of the presentation has been displayed. When the control passes to a state, it means all the events before this state are finished. A state node is a breaking point for two different events. For example, in a presentation sequence, if any media stream is changed, then a new state node is created to distinguish these two events. In our design, when any media stream begins or ends, a new state node is created and an arc connects this new state to the previous state. Therefore, a state node is useful to separate different media stream combinations into different time intervals.

In MATNs, the arc symbol in the outgoing arc for each state will be analyzed immediately so that the process can continue and does not need to know the past history of the presentation. Two state nodes are connected by an arc. The arc labels indicate which media streams are involved. Each arc represents a time interval. For example, when an arc label contains several media streams, then it means that these media streams will be displayed at this time interval. An example MATN will be discussed in Section 4.

3.3. Multimedia Input Strings: Inputs for MATNs

Basically, an MATN is used for the analysis of natural language sentences. Its input is a sentence composed of words. This input format is not suitable to represent a multimedia presentation since several media streams need to be displayed at the same time, to be overlapped, to be seen repeatedly, etc. For this purpose, multimedia input strings are used. Multimedia input strings adopt the notations from regular expressions [11]. Regular expressions are useful descriptors of patterns such as tokens used in a programming language. Regular expressions provide convenient ways of specifying a certain set of strings.

A multimedia input string goes from left to right, which can represent the time sequence of a multimedia presentation. In our design, each arc in an MATN is a string containing one or more media streams displayed at the same time. A media stream is represented by a letter subscripted by some digits. This single letter represents the media stream type and digits are used to denote various media streams of the same media stream type. Two notations $\mathcal{L}$ and $\mathcal{D}$ are used to define multimedia input strings:

- $\mathcal{L}$ = {A, I, T, V} is the set whose members represent the media type, where A, I, T, V denote audio, image, text, and video, respectively.

- $\mathcal{D}$ = {0, 1, ..., 9} is the set consisting of the set of the ten decimal digits.

Definition 1: Each input symbol of a multimedia input string contains one or more media streams which are enclosed by a parentheses and are displayed at the same time interval. A media stream is a string which begins with a letter in $\mathcal{L}$ subscripted by a string of digits in $\mathcal{D}$. For example, V_1 represents a video media stream and its identification number is one.

- **Concurrent:** The symbol "&" between two media streams indicates these two media streams are displayed concurrently. For example, $(T_1\&V_1)$ represents T_1 and V_1 being displayed concurrently.

- **Contiguous:** Input symbols which are concatenated together are used to represent a multimedia presentation sequence and to form a multimedia input string. Input symbols are displayed from left to right across time sequentially. *ab* is the multimedia input string of *a* concatenated with *b* such that *b* will be displayed after *a* is displayed. For example, $(A_1\&T_1)(A_2\&T_2)$ consists of two input symbols $(A_1\&T_1)$ and $(A_2\&T_2)$. These two input symbols are concatenated together to show that the first input symbol $(A_1\&T_1)$ is displayed before the second input symbol $(A_2\&T_2)$.

Hence, a multimedia input string consists of one or more media streams and is used as an input for an MATN.

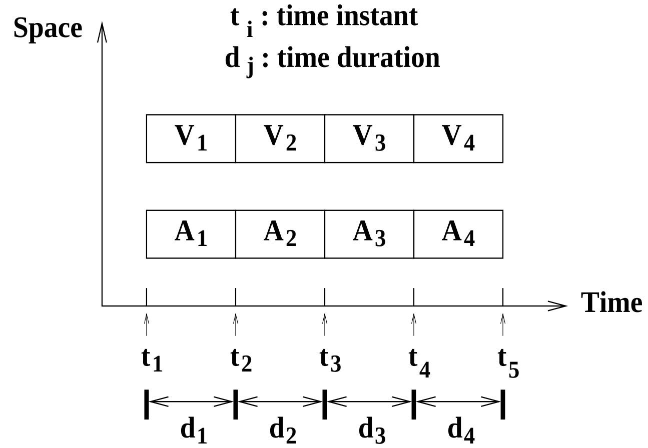

Figure 5. An example timeline for the video and audio streams in the *JTViewer* in Figuer 2.

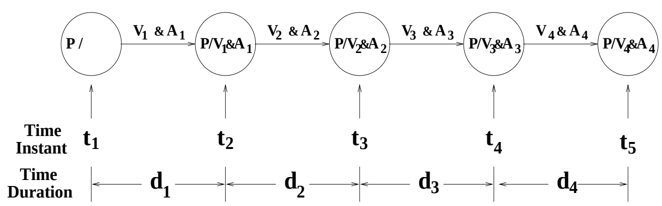

Figure 6. The corresponding MATN for the timeline in Figure 5.

4. Using MATN to Model JTViewer in IP/TV Environments

The MATN is used to model the multimedia presentations for the *JTViewer* in IP/TV environments since different media streams need to be synchronized and displayed in real-time in the *JTViewer*. Figure 5 gives an example timeline model for the video and audio streams in the *JTViewer* system in Figure 2. Since the goal is to provide a live TV-quality distance learning system, this example can be used to represent a TV-like program to illustrate how the multimedia presentation system is modeled by the underlying MATN model. Here, t_i represents a time instant and d_j indicates the time duration for the presentation. In this example, media streams V_1 (video stream 1) and A_1 (audio stream 1) start at time t_1 and play concurrently. As soon as V_1 and A_1 end at time t_2, V_2 and A_2 start to display and end at time t_3. V_3 and A_3 then follow V_2 and A_2 and end at time t_4. At time t_4, V_4 and A_4 start to display and end at time t_5.

The corresponding MATN for the example timeline (as shown in Figure 5) is presented in Figure 6. As mentioned earlier, when any media stream begins or ends, a new state is created and an arc that connects this new state to the previous state is also created in MATNs. In other words, a state node is used to separate different media stream combinations into different time intervals. In this example, there are five state nodes and four arcs which represent five time instants (t_1 to t_5) and four time durations (d_1 to d_4), respectively (as shown in Figure 6).

In Figure 6, the state name $P/$ indicates the presentation P is displayed. The outgoing arc symbol is $V_1\&A_1$ which means there are two media streams V_1 and A_1 starting to display. The "&" symbol is used to represent that V_1 and A_1 are displayed concurrently. The state name $P/V_1\&A_1$ indicates that V_1 and A_1 have been read in presentation P. The next outgoing arc with arc symbol $V_2\&A_2$ is the arc to be followed. This arc consists of media streams V_2 and A_2 that are displayed concurrently. Similarly, the state name $P/V_2\&A_2$ denotes that V_1 and A_1 have been read in presentation P. Then the outgoing arc symbol $V_3\&A_3$ means that V_3 and A_3 display concurrently. The same steps can be applied for V_4 and A_4 until the state name $P/V_4\&A_4$ is reached, which indicates that V_4 and A_4 have been read for presentation P.

The multimedia input string that is the input for the MATN in Figure 6 is shown as follows.

$$(V_1\&A_1)(V_2\&A_2)(V_3\&A_3)(V_4\&A_4).$$

As can be seen from this multimedia input string, there are four input symbols $(V_1\&A_1)$, $(V_2\&A_2)$, $(V_3\&A_3)$, and $(V_4\&A_4)$ to represent the four time durations for the multimedia presentation. In addition, these four input symbols are concatenated together to show that the first input symbol $(V_1\&A_1)$ is displayed before the second input symbol $(V_2\&A_2)$, and then followed by $(V_3\&A_3)$ and $(V_4\&A_4)$.

In this paper, multimedia input string serves as an input for an MATN. The MATN reads each input symbol of its multimedia input string and advances to the next state node to control the multimedia presentation sequence.

5. Conclusion

In this paper, we presented a semantic model called the *multimedia augmented transition network (MATN)* that serves as the underlying semantic model for the multimedia distance learning presentations in the *JTViewer* system. The *JTViewer* system is a Java-centric TV-quality distance learning system that provides TV-quality courseware by taking Cisco IP/TV approach. Both the WWW and the Java technology are incorporated in the *JTViewer* system, which makes the universal accessibility to diverse distance learning services possible. In particular, by applying the Java technology, the cross-platform capability for CISCO IP/TV system can be provided.

In such a TV-quality distance learning system, different media streams need to be synchronized and displayed in real-time. Therfore, the development of a good abstract semantic model for the system is very crucial. There are two main requirements for an abstract semantic model. First, it should be able to effectively provide visualization of control structure for multimedia systems. Second, it should be a good programming data structure for implementation to control multimedia systems. The MATN model meets both the requirements for serving as an abstract semantic model for the mulitmedia presentations in the proposed *JTViewer* distance learning system since it provides a simple visualization control structure and a good programming data structure for implementation of the multimedia distance learning presentation system. A detailed example that illustrates how to use the MATN model and its multimedia input string to model the live TV-like multimedia presentations is also included in this paper.

The *JTViewer* system has been implemented at NKFUST and offered on-demand diverse services and scheduled video delivery. As far as the references surveyed, our work presented is the very pioneering study in the literature towards integrating IP/TV and Java JMF in distance learning applications. Furthermore, the system can be directly ported to any Java JMF-compliant platform for ensuring the goal of architecture neutrality.

6. References

[1] F. Arcelli, M. De Santo, and A. Chianese, "Client-Server Architectures for Distributed Learning Environments: A Proposal," *Proc. of the Thirty-First Hawaii International Conference on System Sciences*, Vol. 7, 1998, pp. 395-403.

[2] S.-C. Chen and R. L. Kashyap, "Temporal and Spatial Semantic Models for Multimedia Presentations," *1997 International Symposium on Multimedia Information Processing*, pp. 441-446, December 1997.

[3] S.-C. Chen and R. L. Kashyap, "A Spatio-Temporal Semantic Model for Multimedia Presentations and Multimedia Database Systems, *IEEE Transactions on Knowledge and Data Engineering*, accepted for publication, 2000.

[4] S.-C. Chen, S. Sista, M.-L. Shyu, and R. L. Kashyap, "Augmented Transition Networks as Video Browsing Models for Multimedia Databases and Multimedia Information Systems," *11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'99)*, pp. 175-182, November 9-11, 1999, Chicago, IL, U.S.A.

[5] Cisco IP/TV Content Manager Version 2.0, 1998.

[6] S. E. Deering and D. R. Cheriton, "Multicast Routing in Datagram Internetworks and Extended LANs," *ACM Transactions on Computer Systems*, Vol. 8, No. 2, pp. 85-110, 1990.

[7] A. Di Stefano, F. Fazzino, L. L. Bello, and O. Mirabella, "Virtual Lab: a Java Application for Distance Learning," *Proc. of the 6th International Conference on Emerging Technologies and Factory Automation Proceedings*, 1997, pp. 93-98.

[8] The IEEE/IEE Electronic Library Online, http://www.ieee.org/ielonline.

[9] R. Jain, "The Convergence of PCs and TV," *IEEE Multimedia*, Vol. 6, No., 4, pp. 1-1, 1999.

[10] Java JMF API, http://www.javasoft.com/products/java-media/jmf/index.html.

[11] S.C. Kleene, "Representation of Events in Nerve Nets and Finite Automata, Automata Studies," *Princeton University Press*, Princeton, N.J., pp. 3-41, 1956.

[12] G. Lawton, "Video Streams into the Mainstream," *IEEE Computer*, July 2000, pp. 12-17.

[13] S.-T. Li, W.-S. Chen and S.-W. Chou, "A Java-based Integrated Asynchronous Distance Learning Paradigm," *Proc. of The First International Workshop on Software Engineering and Multimedia Applications in conjunction with The 11th International Conference on Systems Research, Informatics and Cybernetice*, Baden-Baden, Germany, 1999.

[14] S.-T. Li, T.-S. Li and S.-W. Chou, "JTViewer: A Java JMF-based Viewer for Real-time TV-like Streams," *Proc. of International Workshop on Internet 2000 in conjunction with The 20th IEEE International Conference on Distributed Computing Systems (ICDCS2000)*, Taiwan, 2000.

[15] D. Min, E. Choi, Y.-T. Han, D. Hwang and J.-H. Jung, "A Distributed Multimedia Conferencing System for Distance Learning," *Proc. of Multimedia Software Engineering*, 1998, pp. 88-95.

[16] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. *Object-Oriented Modeling and Design*, Prentice-Hall, Pub. Co., 1991.

[17] E. K. Patterson, D. Wu, and J. N. Gowdy, "Multi-platform CBI Tools using Linux and Java-Based Solutions for Distance Learning," *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 1909-1912.

[18] Bob Peyser, "Which Streaming Media Tool Do I Need?" http://www.streamingmediaworld.com/gen/tools/ipw/, 2000.

[19] RealPlayer for Java, http://www.real.com/.

[20] J. A. Reed and A. A. Afjeh, "Developing Interactive Educational Engineering Software for the World Wide Web with Java," *Computers Education*, Vol. 30, No. 3/4, 1998, pp. 183-194.

[21] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, "RTP: A Transport Protocol for Real-time Applications," RFC-1889, Audio-Video Transport Working Group, Internet Engineering Task Force, Jan. 1996, ftp://ftp.isi.edu/in-notes/rfc1889.txt.

[22] W. Woods, "Transition Network Grammars for Natural Language Analysis," *Comm. of the ACM*, **13**, October 1970, pp. 591-602.