Support For Multimedia On The Web

By
Saurabh Maitra
Department of Computer Science, University of Texas at Arlington

Abstract

Distributed multimedia applications are currently being supported by a growing number of services. However, characteristic requirements of these applications such as higher bandwidth, bounded transmission delay, channel synchronization etc. have led to the need for efficient protocols and schemes used at all layers of the protocol stack from the network to the application layer. An attempt has been made in the paper to analyze the communication requirements of distributed multimedia applications, current networking technologies and communication protocols that support multimedia transmission and finally, software frameworks that support application development and presentation of multimedia content on the web.

1 Introduction

The advances made in the area of local-area networks like Ethernet and Token ring and high-speed networks like Fiber Distributed Data Interface (FDDI) have ensured that most of today's communication needs are satisfied. One area, however, that is causing a major impact on software, hardware structures as well as on communication systems is that of distributed multimedia applications. Examples of such applications are interactive video conferencing, distributed virtual environments, networked computer games and so on. The paper is organized as follows. Section 2 discusses the characteristic features of most multimedia applications. Section 3 delves into a survey of current networking technologies that support multimedia transmissions. Section 4 focuses on the communication requirements of multimedia applications and makes a comparative study of the architecture of two commonly used network layer protocols, namely, Internet Stream Protocol (ST-II) and Resource Reservation Protocol (RSVP). Section 5 deals with three widely used transport layer protocols Heidelberg Transport Protocol (HeiTP), Xpress Transport Protocol (XTP) and Real-time Transport Protocol (RTP). Section 6 describes the support provided to multimedia applications at the application layer. This covers multimedia development microworlds, multimedia frameworks and a standard for multimedia markup language named SMIL. Section 7 presents inference and conclusions.

2 Characteristic Features of Multimedia applications

2.1 High bandwidth requirements

Over the last decade, networking experts believed that multimedia communication would drive bandwidth requirements in the future. Digital video transmission nearly 140 Mbps was considered an important future networking application [1]. To reduce transmission bandwidth requirements, today's systems handle multimedia data mostly compressed formats [1]. The compressed video file format standards currently followed are

a. International Standards Organization (ISO)
b. Moving Picture Experts Group (MPEG)
c. Intel's Digital Video Interactive (DVI)
d. International Telecommunications Union (ITU) H.261

Assuming simplex or unidirectional streams, current bandwidth demand ranges from 0.4 to 1.4 Mbps.

2.2 Transmission Delay

Transmission delay restrictions imposed by interactive distributed multimedia systems are harder than the bandwidth requirements. For instance, consider a telephone conversation via satellite links. Round trip transmission times reach as high as 0.6 seconds making it difficult for the partners to hold a normal conversation. ITU standards specify the maximum total end-to-end delay 150 ms for interactive video applications. This end-to-end delay can be broken down into the following components

a. Source Compression and Packetization delay
b. Transmission delay
c. End-system queuing and synchronization delay
d. Sink decompression, de-packetization and output delay

2.3 Reliability

Traditional data communication ensures reliable end-to-end communication using negative acknowledgements or positive acknowledgement with packet transmission handshake for error recovery. Subsequent re-transmission handshakes can lead to more than a full round trip delay. For time-critical data, retransmitted messages could be rendered useless. Hence, error control and recovery schemes are left to the higher communication layers.

2.4 Channel Synchronization

When audio, video and other data streams come from different sources via different routes, mechanisms are required to achieve the equivalent of lip synchronization [1]. This can be achieved by using a combination of time-stamping and playout buffers.

3 Survey of Network Architectures supporting Multimedia applications

3.1 LAN Architectures

Ethernet

Since Ethernet offers a bandwidth unto 10 Mbps, it allows for four parallel compressed video streams. Currently, many multimedia applications use Ethernet as their transport mechanism although in a controlled and protected environment such as limited number of stations. A typical case where Ethernet can be used is, say, three active stations participating in a DVI-based conferencing application where bandwidth contention is not a serious issue. It does not fare well for distributed multimedia because of the following reasons

1. Non-deterministic behavior of the CSMA-CD protocol used. No control over access delay or available bandwidth per application
2. No provision for access-priority mechanisms. Real-time traffic cannot be preferred over conventional data transmissions when required.
3. In practice, an Ethernet adapter can manage only a limited number of multicast addresses.

Isochronous Ethernet

Isochronous Ethernet can be considered as a local area ISDN extension on an existing networking base (Ethernet). Its ISDN-like channel structure is designed for audio and H.261-coded video transmissions. However, bandwidth and multicasting support limitations make it tough to run DVI and MPEG coded streams.

Token Ring

The total available bandwidth in token ring is nearly 16 Mbps, which proves to be enough for a limited for a number of multimedia streams. . Further, MAC-level priorities are supported thus enabling preferential treatment to real-time data. The presence of priority access and bandwidth management schemes can provide access delay guarantees. Since it also offers multicasting, it can be concluded that token ring is a viable network for multimedia communication.

100 Base-T

Standardized by the IEEE 802.3 working group, 100 Base-T was initially designed to scale the Ethernet approach to 100 Mbps. It is a proposal by 3Com, Cablettron and other companies. Since, it inherits most of the features from Ethernet, it shares the same limitations with regard to access delay characteristics. Although, available bandwidth is much higher, usage rarely exceeds 50 % of the maximum bandwidth. Multicasting support is available to a certain extent. Henceforth, it is not a very good alternative for running multimedia applications.

Demand Priority LAN

This is another 100 Mbps LAN technology being standardized. It was originally proposed as 100 Base-VG by Hewlett-Packard, AT&T and IBM. It is evolution of standard Ethernet and Token Ring to 100 Mbps over voice-Grade cabling. Its main goal is to increase bandwidth while protecting existing interconnection investments. This star-type architecture uses frame switching based on a round-robin access control scheme. This scheme can guarantee delay bounds for any maximum packet size and umber of attached stations. A two-level priority scheme that favors multimedia over regular traffic can further reduce the delay bounds. Multicasting support is also inherited from Ethernet and token ring structures. To sum it up, it is a better architecture for multimedia applications than 100 Base-T and is a viable alternative for a small sized topology.

3.2 MAN Architectures

FDDI

The Fiber Distributed Data Interface can be thought of as a large-scale fast token ring network. The larger bandwidth offered here supports a larger number of multimedia stations. In addition to priority traffic, FDDI also supports a synchronous traffic class, allowing the delay upper bound to be configurable at ring initialization time. This feature in addition to available multicasting makes FDDI support multimedia communication well.

DQDB

Current services are, in general, restricted to connectionless asynchronous traffic. The delay behavior will be suitable for transmitting multimedia only if the MAN remains lightly loaded. Also, multicasting over DQDB is understood to be difficult. Although, the media-access protocol used by DQDB is claimed to have support for connection-oriented multimedia traffic, the IEEE 802.6 working group is defining a framework for connection-oriented services with throughput and delay guarantees.

3.3 WAN and other Architectures

ISDN

The Integrated Services Digital Network is a WAN architecture that was designed to support a large variety of services, from data over voice to fax and video. Due to the absence of multimedia communication requirements in the designs of X.25 or Frame-relay services, ISDN appears to be the only available choice for interactive wide-area multimedia communications aside from leased-line services. It is widely used in Europe. ISDN possesses the following advantages over other WANs for multimedia communication

a. Wide availability

b. Isochronous characteristics
c. It is built on synchronous 64-Kbps channels, which can be used for continued connection-oriented video transmission as well as packetized communication.

Its limitation lies in the relatively limited bandwidth. This restricts the packet size to ensure that the delay incurred is at an acceptable level. Further, lack of multicast services restricts it to point-to-point rather than multiparty conferencing environments.

ATM

The ATM Forum initially defined the use of the 100-Mbps FDDI interface for ATM-based LANs. Even 25Mbps of dedicated bandwidth far exceeds the 1.4Mbps target found practical for multimedia transmissions. It is projected that ATM interfaces will grow to support bandwidth up to 2.4 Gbps. The architecture provides high-speed transmission links and fast switches as a result of which latency in cell transmission is grossly reduced. Buffering delays can also be reduced by interleaving ATM cells. The presence of a signaling protocol allows the building of multicast trees. However, the secret to the high performance of ATM lies in the presence of the ATM Adaptation Layer (AAL) which bridges the gap between the application requirements (which may communicate using variable sized packets) and the underlying ATM layer using fixed-sized cells.

The International Telecommunications Union specifies four service classes classifying different types of traffic. They are as under

a. Class A – constant-rate synchronous bit streams
b. Class B – variable-rate compressed audio/video streams
c. Class C – existing connection-oriented communication services
d. Class D – existing connectionless communication services

In order to support these different classes of service, the following adaptation layers were proposed by the ITU and the ATM Forum. They are [1]

a. AAL 1 – supports Class A services
b. AAL 2 – supports Class B services. Implementation of this layer has been said to be difficult because of the variable rate traffic, reserving peak bandwidth and ensuring guarantees becomes tough.
c. AAL 3 / 4 – provides Class C and D services. The implementation allows interleaved transport of different messages on the same virtual channel using a message identifier (MID) field.
d. AAL 5 – Initially known as Simple and Efficient Adaptation Layer (SEAL), this supports the service classes C and D. It differs from the 3 / 4 layer in that it does not support interleaving and hence reduces the cell segmentation overhead for larger messages and utilizes the available bandwidth better. However, the limitation suffered is that corrupted cells need to be discarded unlike the previous layer where means are provided to localization of bit errors and partially correct messages can still be used by certain media streams.
e. AAL 6 – So far, there are no clear directions for AAL 6. the ATM forum had initially proposed this layer to suit packetized multimedia streams, particularly for MPEG and MPEG – II video.

From the perspective of multimedia communication, the choice trims down to AAL 3 / 4 and 5 layers. AAL 5 has a widespread implementation whereas AAL 3 / 4 has error handling mechanisms that can prove to be useful. To summarize, ATM is the best match for requirements of multimedia applications as it provides more than enough bandwidth, delay and jitter characteristics are bounded under normal network load and multicast communication is provided.

4 Communication Services for Multimedia

The two aspects that exceed the communication services in the network layer level for multimedia from that normally followed are

4.1 QoS support

Audio/Video transmissions require a certain level of service, in other words QoS, guarantees throughout the transmission. Mechanisms are to be provided for

i) Allocation/ de-allocation of appropriately configured channels
ii) QoS level negotiation between end systems, intermediate systems
iii) Control of the QoS level agreed upon

4.2 Group communication support

Multimedia communication often involves groups of more than two users. The key aspects that need to be analyzed are

a. Static/dynamic nature of membership during user participation
b. Centralized/distributed control of membership
c. Homogenous/heterogeneous characteristics and requirements of each member

4.3 Communication Architecture

The motivation for new protocol suites has been a result of the following differences in multimedia communications when compared to traditional data transmissions [1][3][8]

a. Connection management and control for multimedia communication is far more complicated than in a TCP/OSI protocol stack.
b. The communication scheme after channel establishment is much simpler for multimedia applications. Error and flow control requirements are less stringent than that in reliable data communication as offered by, say, TCP.
c. The control communication might have different QoS requirements as compared to the actual media transfer.

Two network layer protocols that provide elaborate mechanisms for connection control and have found widespread acceptance are briefly discussed below

4.3.1 ST- II

The experimental version of the Internet Stream Protocol consisted of a signaling component called Stream Control and Message Protocol (SCMP) and a lightweight data transmission protocol [5]. ST II has been implemented over various platforms like DEC, HP, IBM, Siemens/SNI and Sun. A revised version of the ST II, RFC 1819 [6], was later made available in 1995 that incorporated experiences from these various implementations.

4.3.2 RSVP

The design principles behind the proposal for Resource Reservation Protocol are given as under [4]

1. Receiver-initiated reservation

Receivers choose the level of resources reserved and are responsible for initiating and keeping the reservation active as long as they want to receive data

2. Separating reservation from packet filtering

The functionality of resource reservation and the decision to select packets that can use those resources are separated. The latter is also termed Packet Filter.

3. Provision for different reservation styles

Having three different reservation styles viz., no filter, fixed-filter and dynamic-filter allows intermediate switches on a path to efficiently merge individual reservation requests for the same multicast group

4. Maintaining "soft state" in the network

Maintaining certain state information at the intermediate switches and leaving the reservation responsibility to the end users helps adjust resource reservations in a dynamically changing network.

5. Protocol overhead control

The three overhead factors in the protocol are the number of RSVP messages sent, their size and the refresh frequencies of the path and the reservation messages. These overheads are controlled by merging of path messages and tuning the timeout values carried in path and reservation messages.

6. Modularity

Modularity in the protocol is introduced by ensuring that few assumptions are made with regard to the three components it interface with viz., the QoS parameters also known as flowspec, the network routing protocol and the network admission protocol.

4.3.3 Comparison

The major difference between the protocols is their position with respect to the protocol stack [7]. ST-II is an independent network layer protocol in that it replaces the need for IP. RSVP, on the other hand, works in tandem with IP controlling the way in which packets are transmitted. Although, the protocols discussed above perform well with regard to QoS support, RSVP provides better group communication support owing to its receiver-initiated connection strategy and provision for different resource reservation styles. While ST-II assumes a homogenous multicast tree, RSVP provides support for heterogeneous receivers and trees. This implies that in a multicast group where n participants are directly connected to a high-speed network like ATM and k other participants are indirectly connected through a slower network like ISDN, using RSVP, the n receivers can enjoy the full QoS and the k receivers will be allocated a reduced QoS whereas under ST-II, all the receivers will receive the reduced stream.

5 Transport Layer Protocols

5.1 Design Goals

The need for efficient transport layer protocol for multimedia communication arises from the fact that flow control, error control and synchronization aspects are not covered by the underlying layer network protocol to ensure lower transmissions delays. Further, if the network layer does not support group management, then multicast decisions need to be made at the transport layer.

5.2 Heidelberg Transport Protocol (HeiTP) [9]

HeiTP was designed over ST-II to provide the functionality mentioned above. HeiTP supports connection-oriented communication only. Connections can be released either by the target or the origin. It adds multicasting functionality by the concept of complete and partial connections. A Complete connection specifies all the targets to be specified at connection establishment time whereas a partial connection allows targets to join/leave the connection at their description. HeiTP provides flow control by adopting a pure rate control scheme where the receiver is never forced to send the control information. With respect to data corruption, options are provided wherein corrupted data could be detected, discarded and retransmission requested or passed to the application after flagging it. Late data, that could be useless for the multimedia application, is detected by measuring the elapsed time between two consecutive packet receptions and checking it against the logical gap between them.

5.3 Xpress Transport Protocol (XTP) [10]

Work leading to XTP began in late 1986 at Bell Labs. The primary design goal of XTP was a protocol design that would facilitate the design of supporting VLSI and would be suitable for operation at gigabit speeds. Several service models like message, stream, transaction and multicast were to be supported by a single mechanism. Error control in XTP was initially a Go-back-N protocol. Subsequently, a selective re-transmission method has been included into the implementation that is based on a sequence number called spans. These sequence numbers represent "*islands of correctly received data*".

5.4 Real-time Transport Protocol (RTP) [11]

RTP is the Internet standard protocol for the transport of real-time data. It is widely used for interactive services as well as media-on-demand although it does not scale well to distributed interactive media. The functioning of RTP relies on resource reservation protocols. It is being used over UDP/IP as well as over AAL 5/ATM. Briefly, the protocol has two components that serve the data and control part of a message. The first component supports the real-time properties like continuous media, loss detection, security and content identification. The other component called Real-time Control Protocol (RTCP) provides support for group management, synchronization of media streams and QoS feedback from receivers to a multicast group. RTP does not fare well for transmission of interactive media. The problem with using RTP for distributed interactive media is to maintain consistency in the shared state for all participants in the session [12]. An application-level protocol called RTP/I which is based on RTP meets the requirements of interactive media

6 Multimedia Application Development

Multimedia applications mainly comprise software that requires programming activity and content that needs authoring. Currently most of the multimedia services do not provide much abstraction to the developer other than the API. Muhlhauser and Gescei, in their paper [2], discuss paradigms to describe abstractions to different services that can be provided to the developer. The following classes of support systems are briefly described below

6.1 Multimedia-augmented services

Two general-purpose computing functionalities that need to be adapted for specific use in distributed multimedia systems are

a. Communication protocols – The requirement for multimedia-capable network and transport layer protocols have been discussed in the earlier sections
b. Operating Systems – the primary requirements of multimedia-capable operating systems are
   - QoS-based resource management
   - Real-time CPU scheduling
   - Complimentary memory management policies to support scheduling
   - Support for real-time synchronization
   - Low overhead of task management

To benefit from these adaptations, application developers must rely on simple APIs.

6.2 Multimedia development microworlds

Microworlds refer to programming toolkits that are usually self-contained and platform-dependent [2]. They abstract the low-level system functions at the API level but the APIs of different microworlds have no compatibility between them. Some of the well-known microworlds are Apple's Quicktime, Microsoft's Windows Multimedia Extensions (MME) and IBM's Multimedia Presentation Manager for OS/2 (MMPM/2). Quicktime was implemented with the idea of integrating continuous media with traditional data types. Its main abstraction is the movie data type that also covers audio and other media. A recent addition to Quicktime has been abstractions for virtual reality and conferencing. MME provides device abstractions and file services whereas MMPM/2 provides an advanced device abstraction that hides the device implementation details from the developer.

6.3 Multimedia frameworks

Current multimedia frameworks offer a certain level of sophistication beyond the features of microworlds [2]. They facilitate multimedia development in heterogeneous environments. A more important design goal here is to make frameworks extensible and portable. Two common frameworks are Sun's Java Multimedia Framework (JMF) and Interactive Media Association's framework, Multimedia System Services (MSS). MSS supports distributed objects by complying with CORBA's Common Object Services Specification (COSS) [18]. The basic classes in MSS include virtual devices and media streams as in JMF. JMF provides a set of interfaces to support the RTP protocol.

6.4 Distributed application development

The distributed object-oriented (DOO) paradigm illustrated in Figure 1 can be extended to multimedia applications due to its simplicity and rigid encapsulation features [2]. In contrast to other models like RPC, DOO supports distribution of fine-grained objects at run-time.

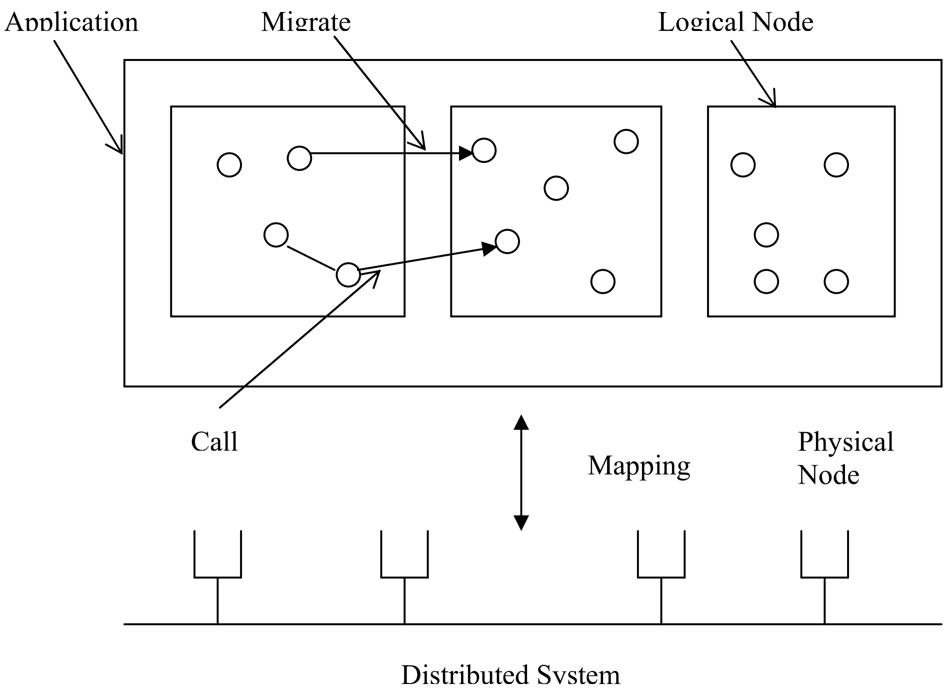

Figure 1. Distributed Object-oriented Paradigm

The hypermedia paradigm is centered around four basic elements – nodes which are meaningful units of information, links which are relations between nodes, anchors – selections within nodes and webs – sets of nodes and links also known as hypertext/hypermedia documents [15].

Figure 2 illustrates how the abstractions required in the context of multimedia services can be integrated considering the above paradigms discussed [2].

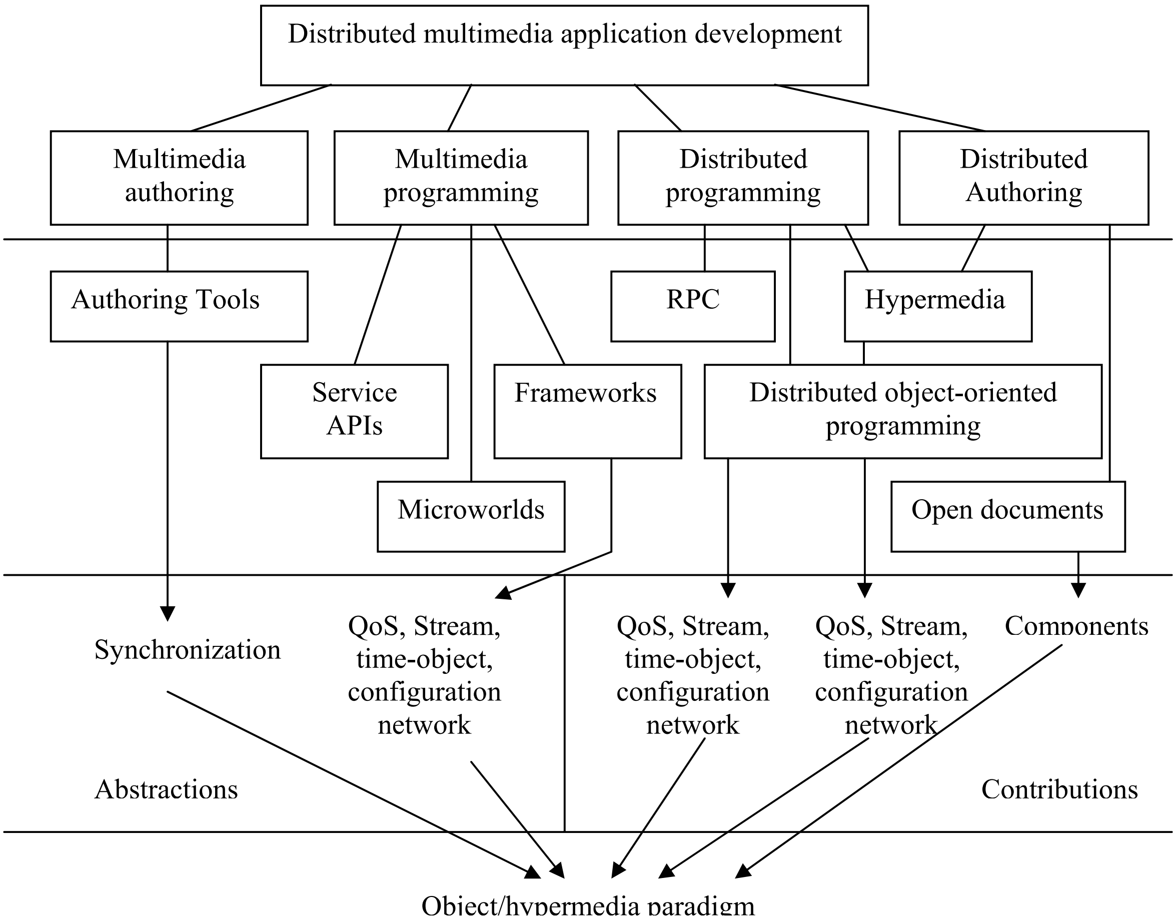

Figure 2. Encompassing paradigm for distributed multimedia applications

6.5 SMIL Overview

Synchronized Multimedia Integration Language is a standard proposed by the World Wide Web Consortium to encode and deliver multimedia presentations over the web. SMIL was initially released in 1998. It is a collection of XML elements and attributes that can be used to describe temporal and spatial coordination of the media objects [14][16][17]. It has been adopted in RealPlayer G2, Quicktime 4.1 and Internet Explorer 5.5. As shown in Figure 3, SMIL 2.0 defines 10 major functional groupings of elements and attributes. Of these, the timing and synchronization group forms the core of the SMIL specification [13].

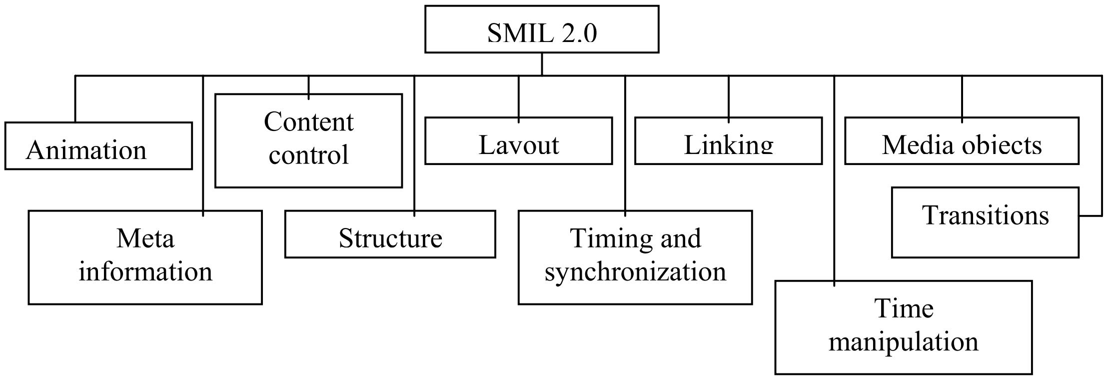

Figure 3. Functional grouping of module sets in SMIL 2.0

7 Conclusion

The support for distributed multimedia communication requires a suitable network, efficient communication protocols providing resource reservation, QoS support, multicast services and flexible error control strategies. Although quite a few network technologies promise support for multimedia traffic, only ATM and ISDN, to a certain extent, have been seen to scale well to the requirements. New protocols like ST-II and RSVP at the network layer and RTP, XTP and HeiTP at the transport layer have been found to support the services discussed above. Current research is more focused on providing a means to support adaptive media streams in a network where bandwidth reservation cannot be provided. Developing multimedia applications initially focused on solutions for stand-alone systems before "going distributed" [2]. With the growth in the support for multimedia in distributed systems, interfacing inconsistencies in the development of such applications has increased as well. An integrated framework of distributed object-oriented system and the hypermedia model can be used to design abstractions for building distributed multimedia applications.

8 References

[1] Heinrich J. Stuttgen, "Network Evolution and Multimedia Communication", IEEE Multimedia, Vol. 2, No. 3, Fall 1995, pp. 42-59

[2] Max Muhlhauser, Jan Gecsei, "Services, Frameworks, and Paradigms for Distributed Multimedia Applications", IEEE Multimedia, Vol. 3, No. 3, Fall 1996, pp. 48-61

[3] Ljiljana Trajkovic, S. Jamaloddin Golestani, "Congestion Control for Multimedia Services", IEEE Network Magazine, Vol. 6, No. 5, September 1992, pp. 20-26

[4] Lixia Zhang et al., "RSVP: A New Resource ReServation Protocol", IEEE Network Magazine, Vol. 7, No. 5, September 1993, pp. 8-18

[5] "Experimental Internet Stream Protocol, Version 2", Internet RFC 1190, C. Topolcic, ed., October 1990, http://www.faqs.org/rfcs/rfc1190.html

[6] "Internet Stream Protocol Version 2 (ST2)", Internet RFC 1819, L. Delgrossi, L. Berger, ed., August 1995, http://www.faqs.org/rfcs/rfc1819.html

[7] L. Delgrossi et al., "Reservation Protocols for Internetworks: A Comparison of ST-II and RSVP", Fourth International Workshop on Network and Operating Systems for Digital Audio and Video, November 1993, Lancaster, UK

[8] Gordon Blair et al., "Towards New Transport Services to Support Distributed Multimedia Applications", Proceedings of the Fourth IEEE COMSOC International Workshop on Multimedia Communications: Multimedia '92, April 1-4 1992

[9] L. Delgrossi et al., "HeiTP – A Transport Protocol for ST-II", Proceedings of the IEEE Globecom 1992, Vol. 3, CS Press, Los Alamitos, California, pp. 1369-1373

[10] G. Chesson, "The Evolution of XTP", Proceedings of the Third International Conference on High Speed Networking, North-Holland, Amsterdam, 1991, pp. 15-24

[11] Henning Schulzrinne et al., "RTP: A Transport Protocol for Real-Time Applications", Internet Draft, February 1999

[12] Martin Mauve et al., "RTP/I – Toward a Common Application Level Protocol for Distributed Interactive Media", IEEE Transactions on Multimedia, Vol. 3, No. 1, March 2001, pp. 152-161

[13] Dick C. A. Bulterman, "Models, Media and Motion: Using the Web to Support Multimedia Documents", Proceedings of the INFWET97, 1997

[14] Lloyd Rutledge, "Multimedia Standards: Building Blocks of the Web", IEEE Multimedia, Vol. 8, No. 3, July-September 2001, pp. 13-15

[15] Michael Bieber, Fabio Vitali, "Toward Support for Hypermedia on the World Wide Web", IEEE Computer, Vol. 30, No. 1, January 1997, pp. 62-70

[16] Dick C. A. Bulterman, "SMIL 2.0 Part1: Overview, Concepts and Structure", Vol. 8, No. 4, October-December 2001, pp. 82-88

[17] Dick C. A. Bulterman, "SMIL 2.0 Part2: Examples and Comparisons", Vol. 9, No. 1, January-March 2002, pp. 74-84

[18] Interactive Multimedia Association, "Multimedia Systems Services", Draft Recommended Practice, 1995