A SURVEY OF NETWORKED AND COMPACT DISC TECHNOLOGIES AND APPLICATIONS FOR INTERACTIVE MUSIC SYSTEMS

BY

SEAN C. TERRIAH

FACULTY OF MUSIC
MCGILL UNIVERSITY, MONTREAL
NOVEMBER 1995

THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES AND RESEARCH
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF ARTS IN COMPUTER APPLICATIONS TO MUSIC

ISBN 0-612-12094-5

TABLE OF CONTENTS

Abstract

The proliferation of the Internet and the convergence of telecommunication, computing, compact disc, and media technologies have paved the infrastructure for new interactive music environments. It is essential for those in the field of new media to be aware of the ramifications of these developments. Furthermore, there has been a growing interest among new media developers to provide the tools necessary to allow an end-user to take an active role in the outcome of an interactive multimedia presentation. This concept introduces the notion of end-user as artist and collaborator. This thesis investigates the diverse applications and technologies that have emerged through cross-disciplinary collaboration and how they may be used to specify and design innovative musical environments.

Résumé

Le développement rapide d'Internet et la convergence de nouvelles technologies reliées aux domaines des télécommunications, de l'informatique, du disque compact et du multimédia ont pavé la voie au développement de nouveaux environnements musicaux interactifs. Une bonne compréhension de ces transformations est essentielle à tous ceux oeuvrant dans le domaine. De plus, il y a un souci grandissant d'impliquer l'utilisateur de systèmes multimédias en lui faisant jouer un rôle actif dans le processus créateur. Cette thèse se propose d'identifier les différentes applications et technologies qui ont émergées d'un processus de collaboration pluridisciplinaire et de démontrer comment elles peuvent être utilisées pour le développement d'environnements musicaux novateurs.

Acknowledgements

I would like to thank Prof. Bruce Pennycook for his support and for having provided a good learning environment and the resources needed to complete this study. I would also like to thank Sean Ferguson and René Quesnel for their help in proof reading this document, and the following graduate students in computer applications to music: Dale Stammen, Jason Vantomme, Kharim Hogan, Ichiro Fujinaga, Paul Kennedy, Kyle Dawkins, and Tang-Chun Li for their advice.

Finally, a special thanks goes to Karine Joly for her unending patience and support.

Chapter One
Introduction

The proliferation of Internet applications and interactive music CD-ROMs has created new perspectives on composing and performing works for new media. The majority of these works are marketed for single-user, single-machine applications. A growing number of releases are intended for networked, multi-user collaboration. This thesis will address both of these issues.

When the infrastructure of the Internet was defined in 1968 it provided a new means of data communication between connected computers. It has subsequently expanded to support applications for videoconferencing (e.g. CU-SeeMe, and the MBone), multi-participant virtual environments (e.g. MUDs, and MOOs), real-time digital media broadcasting (e.g. RealAudio), hypermedia document browsers (e.g. World Wide Web), and interactive media presentations (e.g. Java applets). While these applications significantly extend the Internet, the creation of a networked music environment over a global network has yet to be explored fully.

Compact disc technology has not only re-defined the concept of "mixed media," it has also become the principle format for interactive music environments. The recent trend toward enhancing normal audio-CDs with interactive multimedia content has resulted in new forms of artistic expression for both new media and recording artists. Due to Peter Gabriel's *Xplora1* innovative interactive music CD-ROM, prominent recording artists have become intrigued with the possibilities of this medium to define or distribute their own artistic works. However, other than American composer and electronic music pioneer Morton Subotnick (*All My Hummingbirds Have Alibis*) , no other serious contemporary music composer has written specially for this medium. This may be attributed to the fact that interactivity has been primarily associated with games and education and not with serious art music. Without further speculating on the reasons why art music has not surfaced in CD-ROMs and the Internet, it is nonetheless important to identify what these media have to offer serious composers.

This thesis is an overview of multimedia applications and the ramifications of current networking and CD-ROM technology for innovative music systems.

Chapter Two
Digital Media Fundamentals

2.1 Introduction

It is important to identify the demands of digital media types for networked music systems. Digital media may be either time-dependent (i.e. media that relies on time for playback such as sound) or time-independent (i.e. time is not part of the semantics of the media) and networked systems need to take this difference into account. The type media used in a multimedia presentation ultimately defines the level of user interaction and control over media components. This chapter describes these issues.

2.2 Continuous and Discrete Media

Digital audio, motion video, computer animation, and MIDI (Musical Instrument Digital Interface) are considered to be time-dependent or continuous media. Continuous media requires continuous playout as time passes.[1] Time-dependency among items is part of the information itself, and thus the sense of continuous media becomes altered if the timing, ordering and the speed of the information packets are modified.

Digital audio and motion video contain sequences of coded samples that represent the essential components (amplitude, frequency, resolution, and frame rate) of the information. Computer animation is defined by a sequence of moving objects with time-dependent directives. The speed at which these sequences are played back, effects the semantics of the media. MIDI is a symbolic representation of music (not actual sound) in the form of performance gestures and control information sent between two electronic music devices. It is a time-dependent media that contains time-stamped events (notes, velocity, and control data) in the order in which they were encoded. The time-stamps are then used as directives for the playback of musical data. [1]

In contrast, time is not part of the semantics of text, graphics, and images. Their information units (pixels, characters, objects and lines) are time-independent and they are therefore considered discrete media.

2.3 Time-Dependent Presentations

In time-dependent multimedia presentations where various media components are synchronized between elements of different types, the classification between time-dependent and time-independent media becomes blurred. This is partly because the meaning of the overall presentation depends on media of different types , whether continuous or discrete. These presentations can be realized with time-based authoring

[1] Comparing MIDI and digital audio is analogous to comparing graphics and bitmaps for line drawings. Fluckiger, François. *"Understanding Networked Multimedia Applications and Technology."* Prentice Hall. 1995: p. 587.

systems such as Director, or with programming languages including Visual Basic, C and Java.[2]

Director is a sophisticated multimedia authoring system for PC Windows, and Macintosh computers. At one level, it is a time-based authoring tool using an animation-rooted organizational scheme in which a series of "frames" makes up a "movie." At another level, Director's Lingo scripting language allows the author to exert (possibly time-dependent) control over aspects of the movie's navigational flow and its component media elements.

2.4 End-User Interaction

Interaction between the participant and the media stored and presented by the computer for interactive media presentations is programmed to permit the end-user to control certain aspects of the presentation flow, to modify media , to collaborate in further development of the plot or composition, and to customize settings. The resources available and the level of user control over media components are limited by the data input resources and the digital media used in the presentation.

2.4.1 Input Resources

The interactive controllers that are typically used for computer-controlled interactive media presentations are the mouse, keyboard, and joystick. Although these devices support the commands and interaction needed for a majority of computer games, for interactive multimedia environments where music is a central element to the presentation, these devices limit the end-users ability for self-expression.

Using digital audio to control navigation and other aspects of a presentation is too data intensive and difficult to implement. An alternative is to use MIDI instruments as input devices to trigger events, to coordinate ideas, and to perform in a computer-controlled music system.

[2] Java is a programming language that is similar to C++ created by Sun Microsystems. It will be discussed further in Chapter 4.

2.4.2 Level of User Control

Most multimedia presentations contain more than one media type, for instance text, graphics, images, motion video, computer animation, digital audio, and in rare cases MIDI. As mentioned previously, the level of user control over a particular type of media is limited by the media used. Digital audio is more difficult to modify and transform in real-time than MIDI data for example, while images and graphics of computer animation are easier to alter than the coded samples of motion video.

MIDI data provides explicit information pertaining to a musical performance. As shown in figure 2.1, the stream of MIDI data provides the musical information for electronic music devices to interpret and perform music.[2]

145	60	75	-- play note C4 with a velocity of 75
129	60	0	-- release note C4
145	61	76	-- play note C#4 with a velocity of 76
129	61	76	-- release note C#4
193	25		-- change timbre (preset) of channel 1 to 25
177	7	100	-- change the global volume on channel 1 to 100
178	7	127	-- increase global volume on channel 2 to 127

Figure 2.1: Simple MIDI stream

With MIDI, modifying musical parameters such as tempo in real-time is trivial in comparison to modifying the parameters of a digital audio file. This is due to the fact that MIDI contains instructions whereas audio is the final format.

Digital audio streams contain discrete samples coded in time that are not explicit in their meaning. Thus spectral analysis and pattern recognition are required to determine the pitches and amplitude of a melody.[3] The processing involved is intensive because of the amount of data needed to encode individual samples of an audio signal. The storage requirements for a 1 second audio file is determined by following formula:

sampling rate (SR) x bit-resolution x 1 or 2 (mono or stereo file) = storage requirements

Compact disc quality audio thus requires a storage rate of 176 KBytes per second (44.1kHz x 16-bit x 2), as opposed to the 3 Bytes required to represent a note-on MIDI message.(See Appendix 2.1)

MIDI provides explicit information and requires very little processing. As a result, it offers the end-user more control over the music than does digital audio.

2.5 Discussion

This document is concerned with the requirements needed to support continuous media over a networked system. Real-time data transmission of continuous media requires that the transmission medium used on a network system respect time-dependencies. The communication methods, protocols, and transmission mediums are crucial to multimedia applications. If the data are not delivered in the proper order, the end-system to which the data are being sent, must re-order the bit stream to ensure that the original data are recreated. This issue will be further discussed in Chapter 3.

Chapter Three
Relevant Technologies for Multimedia Applications

3.1 Introduction

This section describes the relevant technologies used in multimedia applications for networked systems and CD-ROM. Multimedia platforms, data communication techniques, transmission mediums, and data compression will be discussed as they relate to music systems.

3.2 Multimedia Platforms

The following paragraphs will identify the essential components of computer and set-top systems for multimedia applications.

3.2.1 Desktop Computers

While computers were originally invented to "crunch numbers," they are now being used as platforms for all types of media production such as: 1) video editing (Adobe Premiere, Avid, Media 100); 2) graphic design, 3-D modelling and computer animation (Adobe Photoshop, Ray Dream Designer, Soft Image); 3) digital audio editing and recording (Deck, Sound Designer, Sonic Solutions); 4) multimedia authoring (HyperCard, Producer, and Director); 5) interactive performance and composition (MAX); 6) sound synthesis (CSound); 7) CD-ROM entertainment and education; and 8) as an interactive "window" to the Internet.

Macintosh and Multimedia Personal Computer (MPCs running the Windows operating system) are the most commonly used computers.[4] These computers come equipped with fast CD-ROM drives, hardware and software support for motion video playback (e.g. QuickTime), software video compression schemes (Cinepak, Indeo, and MPEG-1), CD-quality audio, and General MIDI capabilities.[5] These systems can support complex multimedia presentations.

3.2.2 Set-Top Platforms

In recent years there has been an explosion in the availability of diverse CD-based gaming platforms. These platforms are capable of more sophisticated audio and graphics processing than the primitive cartridge-based systems of the early 1980s, and are comparable in speed to "low-end" desktop computers.

The best selling game machines include: Sony and Philips CD-i multiplayer, Panasonic's FZ-1 R.E.A.L 3DO interactive multiplayer, the Sony PlayStation, Sega Saturn, and the Ultra 64 by Nintendo and SGI (Silicon Graphics).[6]

Except for CD-i, these game machines all support multiple CD-quality audio channels, real-time video decompression playback (MPEG-1 chip), and are capable of computing Iop to 500,000 texture-mapped polygons per second.

While game systems all support the essential components for multimedia presentations, MIDI is generally not supported. There has been some discussion about implementing General MIDI capabilities into these systems, but hardware to support MIDI input has yet to be implemented.

3.2.3 Digital Media Playback: QuickTime and QuickTime VR

QuickTime is a software-based system extension for Macintosh and MPC computers that is used to create and play digital movies. Media including motion video, text, computer animation, audio, MIDI, panoramic views, and sprites (active/moving graphics) can be integrated into a "movie." In a QuickTime movie, media is arranged in tracks and indexed by SMPTE time code. (See Figure 3.1)

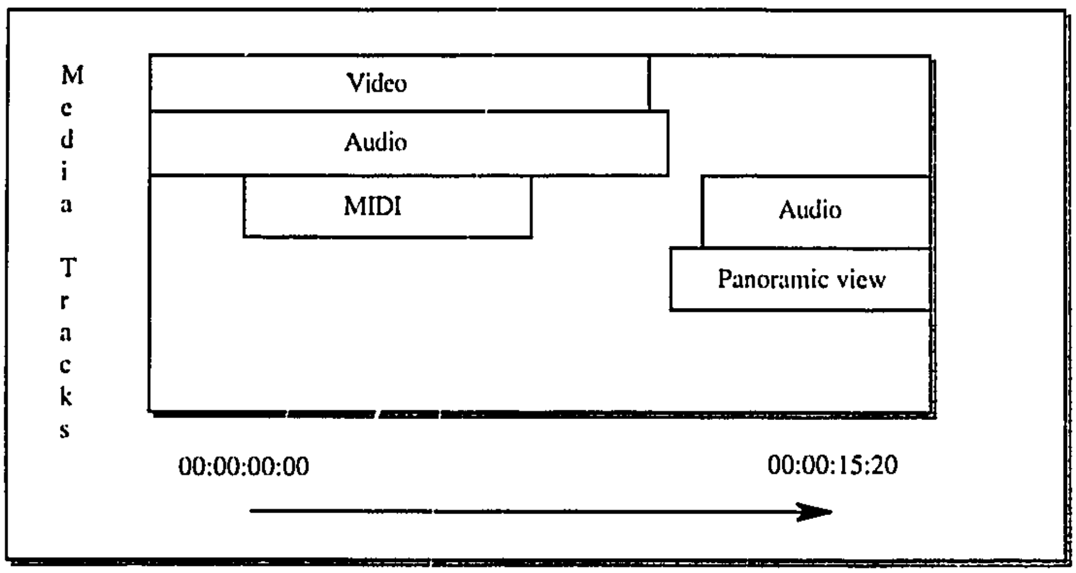

Figure 3.1: Illustrates a QuickTime movie and its interleaved tracks.

QuickTime 2.1 includes an additional track for panoramic views. This event-driven track permits the end-user to navigate in a 3-dimensional (3-D) "image-based" virtual reality environment. The media consists of a series of connected 35-mm photographs used to create QuickTime VR movies. These movies are based on *warping* and *stitching* algorithms that were developed by Apple.[7]

Virtual Reality (VR) involves the immersion of the user into an interactive computer-generated world. Three-dimensional computer-modeled environments are experienced using a head-mounted display (HMD) to interface with a VR system. This device features a real-time stereoscopic display screen, stereo headphones for audio tracking, and a position sensor that provides head-tracking information as input to a host computer. The computer generates perspectives of the environment according to the location of the participants head. Position-sensing gloves such as VPL's Dataglove and Mattel's PowerGlove transmit hand movements to a VR system. These gloves permit users to place their hands "into" a computer-modelled space, allowing them to experience the illusion of "moving," holding", and "playing" objects.

Unlike more complex VR systems (costing at least $100,000), QuickTime VR movies run on low-cost Macintosh and MPCs with no additional hardware. Developers can create graphic "hot-spots" used to activate various actions (audio and MIDI tracks), navigate to other locations, or as a link to another 3-D image-based environment.

3.3 Data Communication

Switching, data transmission, and communication methods are significant issues for real-time multimedia applications being implemented over a networked structure. These networking elements define how media is transmitted between two end-systems.

3.3.1 Networked Switching Techniques

Switching processes ensure that data are properly routed within a networked structure to the appropriate end-system. The two types of switching techniques are circuit and packet-switching.

Circuit-Switching

The telephone network system is based on circuit-switching. This switching technique consists of establishing a physical connection between two network components for the duration of the connection. This serves as an open "pipeline" that permits two end-users to utilize the facility (phone line) within the bandwidth limitations. Telephone lines can support 8 kHz real-time voice messages. To represent this message in the digital domain would require a transmission rate of 64Kbps. Circuit-switch systems provide fixed data rates and can therefore support continous or time-dependent media.

Packet-Switching

In a packet-switching network information is sent as segmented data in the form of individual packets. These packets propagate over a shared connection along with other packets from different systems, all connected by the same wires. Each data packet contains the address of its destination (i.e. like telephone numbers where a unique numerical value is assigned to every device).

The advantage of using a packet-switched network is that it permits concurrent data communication between end-systems. It does not, however, provide guaranteed bit-rates (speed at which bits of information are transmitted), and is prone to improper bit-ordering of individual packets of information. Transmission errors and varied bit-rates can alter the meaning of a time-dependent media. This requires the network hardware of an end-system to re-order the bit stream by applying complex error-correction hardware.

3.3.2 Data Transmission

Transmission methods determine the way in which each individual bit is coded and the timing conditions for sending or receiving sequences of bits. [8] There are three types of data transmission techniques used in data communication networks: asynchronous, synchronous, and isochronous.

In asynchronous transmission, data are sent at irregular intervals with no timing restrictions on when they are to arrive at their destination. By this definition, asynchronous transmission cannot support time-dependent or continuous media without proper bit re-ordering, and is more suited to transmitting time-independent media such as graphics and text.

In contrast, synchronous data transmission relies on a synchronized clocking signal between both end-systems. The transmitting system can use this clocking signal to define a maximum end-to-end delay for packets of a data stream. End-to-end delay is the time required for a packet of information to reach an end-system receiver.

Although the maximum delay time can be set, the minimum delay between data packets is variable. With no set minimum time, packets may arrive too soon. In a situation where CD-quality digital audio (176 KBps) is received 1 second too early, the receiving end-system must store this data in a temporary storage buffer until it is needed. Numerous schemes have been devised to handle these problems.

Isochronous transmission defines both a maximum and minimum end-to-end delay time for each packet. Isochronism is required for a network to satisfactorily transport continous media streams such as real-time audio or motion video.[9] As shown in Figure 3.2, an 8 kHz audio file is digitized at 64 Kbps. To transmit this data in real-time requires that each byte be sent every 125 microseconds. If the data is received too soon (50 microseconds) it needs to be stored in memory on the end-systems computer. By defining a minimum delay time between individual packets, the storage capacity for early data is greatly reduced compared to the potentially large buffers utilized in synchronous systems.

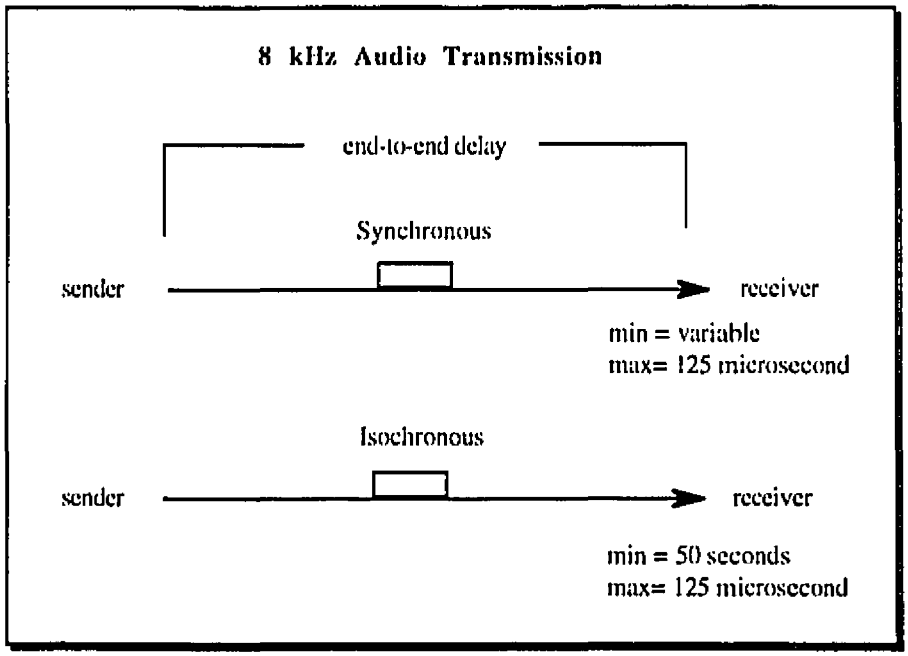

Figure 3.2: Both transmission methods support continous media, however by setting a minimum end-to-end delay time between packets, isochronous transmission requires less storage for data that is received too soon.

3.3.3 Data Communication Methods

There are three methods of transmitting data over a networked structure : 1) unicast; 2) broadcast; and 3) multicast.

As shown in Figure 3.3 these methods have distinct ways of transmitting data.

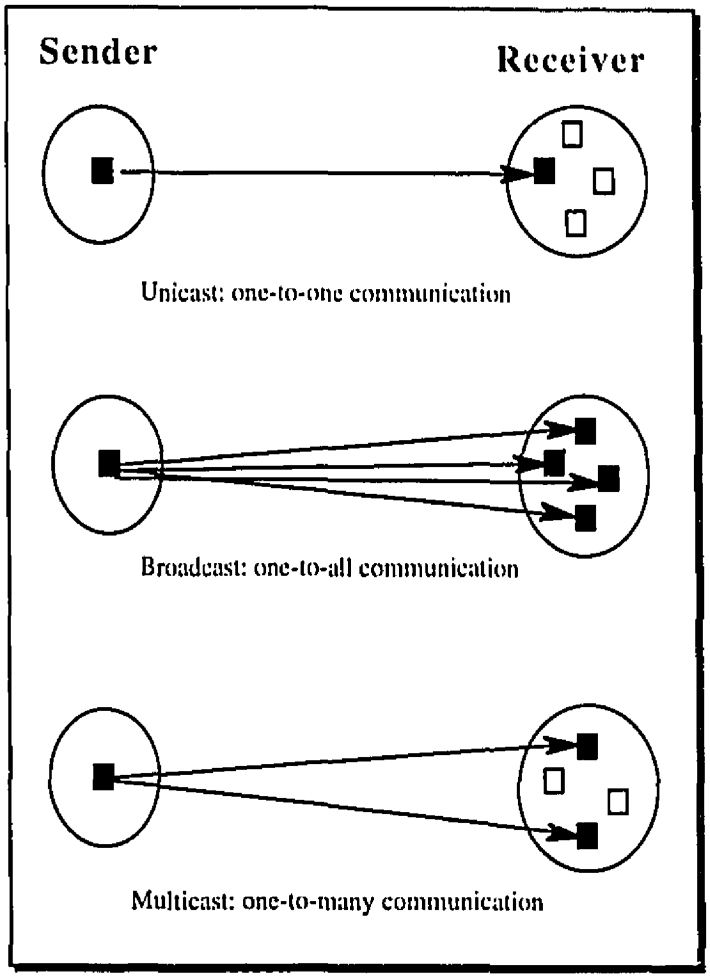

Figure 3.3 Illustrates three types of communication methods

Unicast

In the unicast mode, packets of data are transmitted to a single destination address, and as a result cannot support distributed multimedia content. Unicast communication is by far the most prevalent technique employed by all networks from local to world-wide in scope.

Broadcast

In a Local Area Network (LAN), machines are physically connected by wire. All machines on a LAN receive broadcast packets from a source, even if that packet is not specifically addressed to a particular machine. Broadcast packets are similar to "junk mail," they are sent to all receiving machines over a LAN. Data traffic, in the broadcast mode, is therefore very intensive and poses unacceptable overheads in most cases.

Multicast

Multicast is a form of selected broadcasting. Multicast packets are transmitted to a multicast group where participants can connect to receive packets of information. Data traffic is less intensive using multicast communication, as packets are only sent to those who participate in a multicast session.

Multicast is used for distributed multimedia content over a networked system. Further discussion on multicasting will be provided in Chapter 4, as it relates to the *MBone.*

3.4 Communication Protocols

For two digital devices (i.e., computer, and network hardware) to communicate with one another, there must be a physical connection between them (twisted pair, coaxial cable, and fiber-optics) and an agreement as to the logical organization of the information. Such mutual understanding is usually called a communication protocol and the conduits through which these communication protocols propagate are called transmission mediums. The field of communication protocols is dense and complex and only a general description can be provided within the scope of this document.

3.4.1 MIDI

The standard communication protocol for digital devices designed for musical applications is MIDI. This protocol communicates musical data (performance gestures) and control information between two electronic music devices (See Figure 3.4).

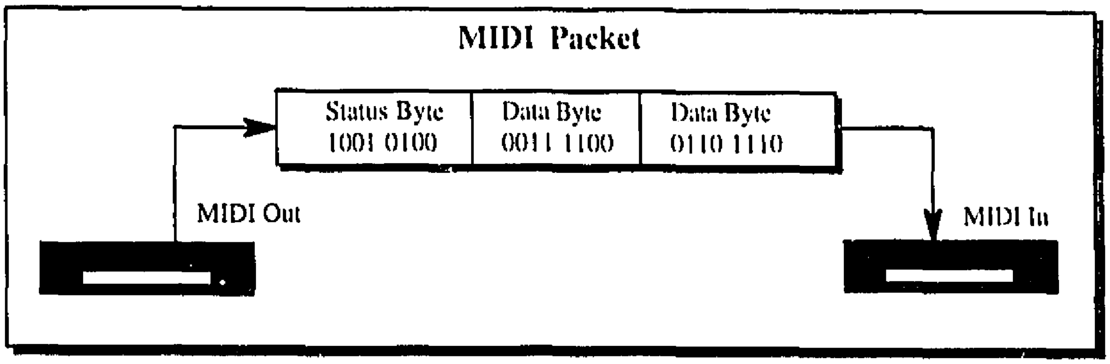

Figure 3.4: This MIDI message instructs the receiving synthesizer to play note 60 with velocity 110 on channel 4

The MIDI protocol is relatively simple and compact. It has been adopted by all electronic music instrument manufacturers.

3.4.2 Internet Protocol Suite

Internet

Computer networking began in 1969 as an initiative of the U.S. Department of Defense to establish a reliable communications network. The result was an effort to link the military establishment, universities, and defense contractors by computer into a network known as the Advanced Research Projects Area Network (ARPAnet).[10] The ARPAnet has exploded into a vast collection of networks called the Internet.

As the Internet emerged, so did the personal computing industry. During this period, there were no set standards for intercommunication between diverse computer platforms, as each system understood different protocols. As a result, there was a need for a common communication protocol that allowed diverse computer systems to exchange digital data with one another. In 1982, ARPAnet defined the standard communication architecture for diverse computer systems connected to the Internet. That standard is commonly referred to as the TCP/IP protocol suite.

The Internet is based on the communication of digital data between networked computer systems. Any media (text, images, video, and audio) represented in the digital domain can exist within and be transmitted across the network of computers that make up the

Internet. This "open system" is profoundly changing the nature of interpersonal communication.

The Internet consists of a large number of interconnected computer networks over a large geographic area. Digital media is exchanged between computer systems over the Internet through a packet-switching network. As a stream of packets propagate over the Internet, routers direct these packets to their proper destination defined by the Internet Protocol (IP) address. Each IP address consists of four sets of numbers separated by periods (e.g. 132.206.44.2). This address is unique and is used by individuals, Internet applications, and computer devices to contact a particular machine.

TCP/IP Protocol Suite

TCP/IP was implemented around the UNIX operating system and consists of four-layers 1) the application layer which is used for Internet applications such as the File Transfer Protocol (FTP) and telnet; 2) the transport layer which offers a host-to-host connection; 3) the internet layer which routes data packets to the IP destination host; and 4) the network access to the physical transmission medium. [11]

3.4.3 OSI Reference Model

The OSI reference model defines standards for linking diverse computer platforms (i.e. other than UNIX), and is a slight derivation from the TCP/IP model in that it consists of seven vertical layers as opposed to four.(See Figure 3.5)

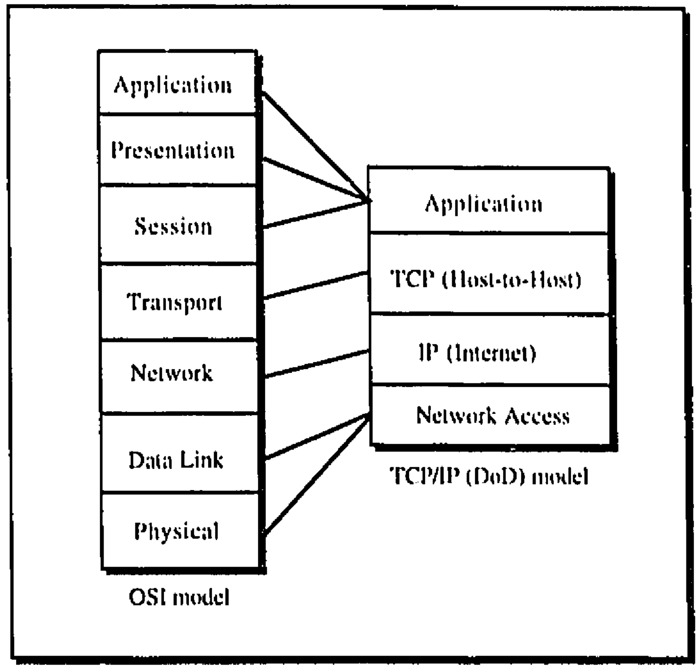

Figure 3.5: TCP/IP is an implementation of OSI where some layers are collapsed.

OSI allows two computer systems conforming to the reference mode and associated standards to connect. As shown in Figure 3.6, OSI defines a basic communication structure between two computer hardware systems.

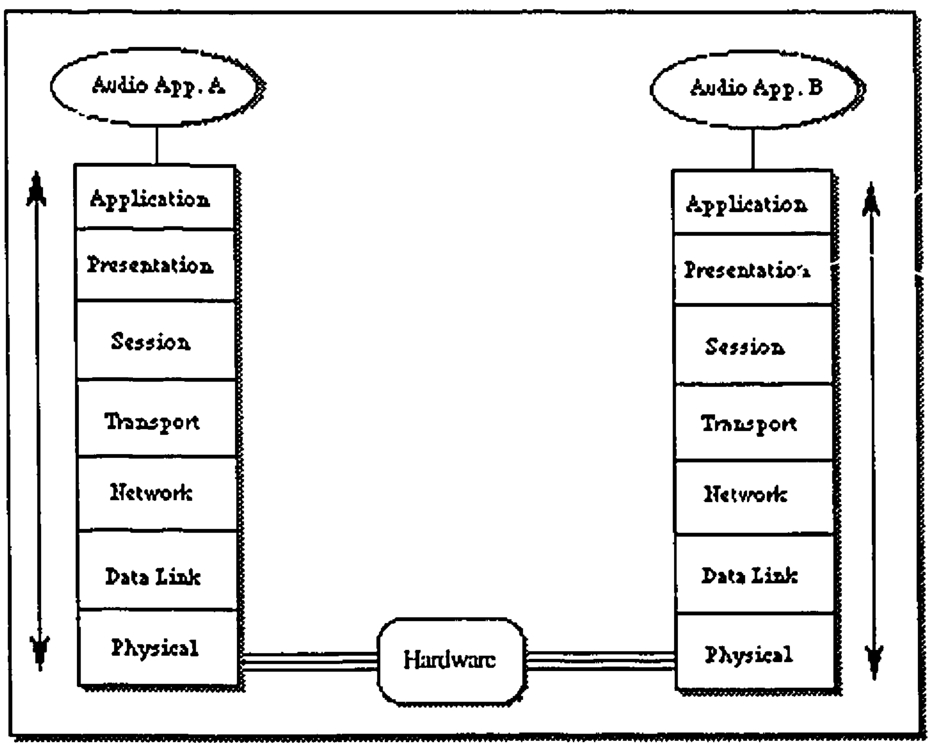

Figure 3.6: Illustrates the seven layers of the OSI reference model

Layers within the OSI stack interact only with their neighbouring "peer" layers (above and below layers), and with their counterpart layer "entities" (same layer at the destination host). Each layer performs a related subset of the functions that are required for intercommunication between computer hardware systems. The required task is divided into a number of more manageable subsets of the original task. Each layer consists of a suite of protocols that are responsible for processing protocol data units (PDU) for peer layers. The protocols add information about the data in the form of *headers* and *footers*, so that receiving computers are able to decode the data packets into their original format. As each layer receives its packet (from its counterpart layer), they strip off the header and then process the data accordingly; passing it on to the next peer layer for further processing. [12]

AppleTalk is a proprietary communication scheme that is based on the OSI reference model. Macintosh computers use MacTCP or OpenTransport (PowerMacs only) to communicate with other systems over the Internet. MacTCP functions as an interpreter between TCP/IP and AppleTalk protocols.

Example Audio Transmission

The following section discusses the processes required for an application to send an audio file to another computer over a networked structure. The application layer (Layer 7) is invoked when two applications on remote computers exchange media; in this case digital audio. The audio file is packaged with a header and sent to its neighbouring peer layer below (Layer 6). The presentation layer receives this data and encodes it into a format that is machine-independent. A presentation header is prepended to the data and passed on to the session layer, which ensures that all pertinent information (destination address, and the size of the data packet) is correct. This information is further packaged into the session layer header and is then sent to the transport layer (Layer 4). The transport layer ensures reliable transmission by storing a copy of the audio data stream. If the TCP protocol is used at this layer, any data packets lost during transmission are retransmitted to the destination host. The transport layer prepends it own header to the data, which is passed on to the network layer (Layer 3). The network layer contains the IP address of the destination host (from the session layer) and determines the fastest route to transmit the data packets according to its routing tables.

The data link layer adds another level of error control to ensure that the data is error free once it reaches the physical layer. The final layer is responsible for transmitting the data over a physical medium. As data packets are processed and passed down from layer to layer, extra information (overhead) is added on to the data. (See Figure 3.7)

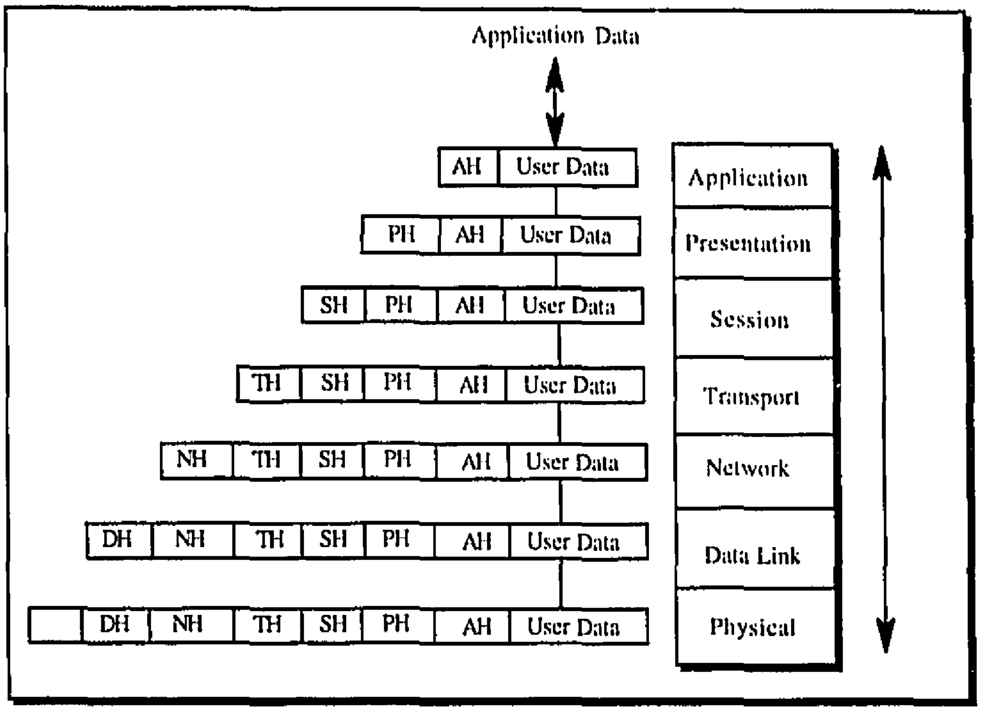

Figure 3.7: As the data passes from one layer to another, headers are prepended to the data. The headers AH (Application Header), PH (Presentation Header), SH (Session Header), TH (Transport Header), NH (Network Header), DH (Data Header), contain the information needed for the receiving system to decode the data packets.

3.4.4 Transport Layer

The transport layer protocols provide the basic end-to-end service of transferring data between users or transport entities. Any process or application can be programmed to access the transport services directly without going through the session and presentation layers.

The TCP/IP and OSI protocol suite both include two transport-level protocols: the connection-oriented Transmission Control Protocol (TCP), and the connectionless User Datagram Protocol (UDP).

The TCP Transport Protocol

As described earlier, TCP is a reliable connection oriented protocol that is used for FTP, and telnet. When a user sets up a file transfer, FTP establishes a TCP connection to the target system for the exchange of control messages. The controls consist of a user ID, password, and specific file commands. When a file transfer is approved, a second TCP connection is set up to transfer the data. The objective of the FTP application is to transfer error free data with no consideration of the speed at which data arrives at the client's computer. This protocol is "connection-oriented" in that a connection must be acknowledged between the source and destination computers; otherwise the data cannot be transmitted. TCP is the most widely used transport protocol of the Internet suite. It is not, however, well suited for real-time multimedia applications, especially for the transport of continuous media.[13]

The UDP Transport Protocol

UDP provides an unreliable connectionless delivery service using IP transport messages between host computers. A datagram transmission is performed without a prior establishment of a connection, nor guarantee of either correct routing or proper ordering of the transmitted data, nor the overhead of the connection-oriented TCP.

UDP datagrams contain the data and the IP destination port of the receiving end-system. The identification port is a "virtual port" negotiated with the receiving computer's operating system for incoming datagrams. If packets are lost, duplicated, delayed, or out of order during a data transmission, the end-system does not acknowledge the fact and the data is not re-transmitted.

Because UDP does not support error detection, data recovery, and the overhead of TCP, it is well suited for certain multicast applications in that high-speed multiuser communication can be achieved. On the other hand, the fact that it is unreliable is a major concern for those wishing to compose interactive works for a networked music system.

3.4.5 World Wide Web Protocols

The World Wide Web (WWW, W3, or Web) is a cooperative project initiated by the Conseil Européan pour La Recherche Nucléaire (CERN) to design and develop a series of concepts, communications protocols, and systems to support the interlinking of various types of information according to the hypermedia concept.[14]

Prior to the Web, data exchange across the Internet was almost entirely text-based. The Web is based on hypermedia (multiple media) "pages" or documents implemented through the Hypertext Markup Language (HTML). HTML is a command language for formatting codes for text, hypertext links to media or other Web pages, and graphics. Hypermedia documents are transmitted between client and server machines using the Hypertext Transfer Protocol (HTTP). An HTTP connection is only active during data transmission. An HTTP transaction between a client and a Web server consist of four stages: 1) the establishment of a connection with the host computer; 2) the sending of a request to the specified Uniform Resource Locator (URL which identifies the document or device); 3) the response by sending the requested media; and 4) the closing of the connection.

A Web browser (application used to view Web documents) connects to an HTTP server specified by a URL. The browser sends an HTTP packet to request a specific item (e.g. URL http://www.mcgill.music.ca/Audio/Beethoven.au) from an HTTP remote server. As shown in Figure 3.8, the server responds by sending the requested media embedded within the Multipurpose Internet Mail Extension (MIME) file format.[15] The client computer receives this MIME file (header and its contents) and determines how to interpret the data. If the response is an audio file, the browser uses "Helper" applications to playback the media. A vast array of these Web helpers are available as shareware from many Web server sites.

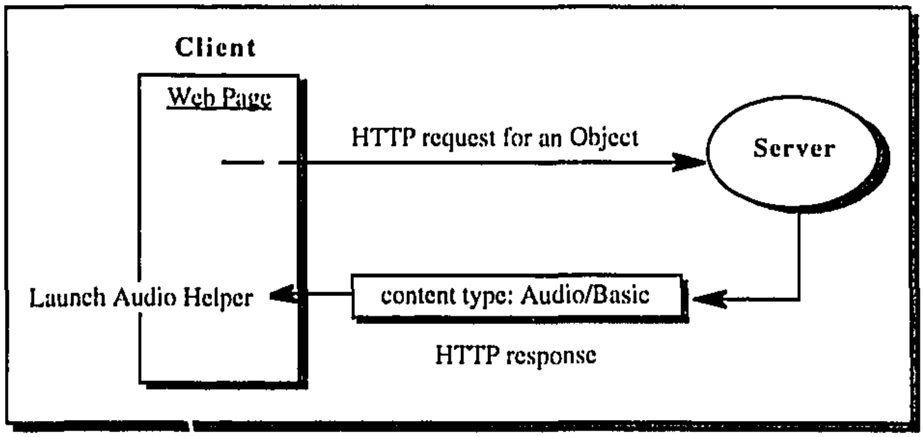

Figure 3.8: Illustrates the flow of events between client and server.

3.5 Transmission Mediums

The topic of network transmission mediums is complex.[1] The speed at which data dissipates over a network is dependent on several factors, including the transmission medium. These mediums can be categorized as circuit-based, packet-based, and CD-ROM.

3.5.1 Circuit-Based Networks

Circuit-based networks consists of a point-to-point connection between two end-systems over a telephone line. The mediums used to transfer data over telephone lines include modems, Integrated Systems Digital Network (ISDN), Broadband-ISDN (B-ISDN), and T-1/T-3 circuits.

Modem

The standard method of transferring data over existing telephone lines is by means of a modem. A typical modem is able to transfer data at approximately 2.4 to 28.8 Kbps across copper twisted-pair telephone lines. A 14.4 Kbps modem is the minimum speed suitable to access hypermedia documents from the Web. At 14.4 Kbps, it would take

[1] Medium is used here to represent the hardware that transmits data (i.e. modem, ISDN, CD-ROM). The plural, mediums, is used to distinguish these technologies from "media".

file (7 Mbits) from a remote machine over the Internet. Applications such as Internet Relay Chat (IRC) and the UNIX "talk" program are able to send text-based data in real-time between two remote machines, but the bandwidth required to send text is many times less than that for digital audio or motion video. It is clear that modems cannot support the transmission of data intensive media for real-time applications.

ISDN

ISDN was designed as the digital upgrade to the existing public switched telephone networking system. It is based on circuit switching of two synchronous bi-directional 64 Kbps voice and data channels (B-Channels) plus an additional signaling channel (D-channel) at 16 Kbps. The total bandwidth of 144 Kbps was developed to carry voice, data, and signalling simultaneously on three separate channels. The basic access connection to an ISDN network is still short of meeting the demands of low quality digital audio (22 kHz 8-bit mono).

B-ISDN

Broadband Integrated Services Digital Network (B-ISDN) is one of the emerging high-speed voice and computer data services for wide-area networks (WAN). Asynchronous Transfer Mode (ATM) is one of several B-ISDN services that provide megabits or even gigabits of bandwidth across a wide-area network. It is a fast hybrid between packet-switched and circuit-switched technology used for multiplexing and transporting voice, data, and video over a cell-switched network at extremely high transfer rates (155 Mbps to 2.2 Mbps).

T-1 and T-3

Until 1991, the main Internet backbone relied on a network of 1.5 Mbps T-1 circuits. The Internet has since been upgraded to a 45 Mbps T-3 circuit. T-1 digital transmission lines provide 24 multiplexed telephone (voice) channels at 64 Kbps per channel, resulting in a total bandwidth of 1.544 Mbps. T-3 supports 672 (28 T-1 circuits) audio channels (64 Kbps per channel) resulting in a total bandwidth capacity of 45 Mbps. T-3 can also support 8 MPEG-2 video channels in real-time.

3.5.3 Packet-Based Networks

Packet-based networks operate over a LAN. The various transmission mediums that exist for LANs include MIDI and networked data transmission mediums such as LocalTalk, Ethernet, and FDDI. These packet-based mediums use the asynchronous transmission method for transferring data between two end-systems.

MIDI

MIDI is a network protocol for electronic music devices. MIDI transfers packets of musical data at 31.25 Kbps. A note-on MIDI message consists of 3 Bytes. As shown in the following formula:

$$31250 / 3 = 1041 \text{ notes per second or } 1 \text{ ms per note}$$
$$10 \text{ notes} \times 1\text{ms} = 10 \text{ ms}$$

it takes approximately 10 ms to transfer a 10 note chord to a receiving MIDI device. Many complex pieces which use MIDI data often exceed MIDIs 31.25 Kbps bandwidth capacity resulting in lost information or MIDI data errors.

Computer Data Transmission Mediums

LocalTalk was the original transmission medium developed by Apple to run AppleTalk protocols between computers and printers. The LocalTalk networking system was designed to run at 230 Kbps.

The Ethernet system is implemented on a LAN, and carries packages of data between computers at a rate of 10 Mbps (10BaseT) or 100 Mbps (fast Ethernet). The most widely used version of the Ethernet technology is the 10-Mbps twisted-pair variety. [16] The 10 Mbps data transmission rate that Ethernet offers represents the total available bandwidth of an Ethernet LAN, and hence degrades with more traffic.

Recently, packet technologies based on fiber optic transmission have been developed. Fiber optics have an extremely high data transmission rate, hence data intensive media such as digital audio and motion video can propagate in real-time. FDDI can sustain data transfer rates of 100 Mbps to 200 Mbps. It is used primarily to interconnect high-speed LANs. [17]

3.5.4 CD-ROM

The speed and the accuracy at which data is retrieved from a CD-ROM is determined by the average data access time, data transfer rates, and buffer size of the CD-ROM drive. These factors are crucial for multimedia CD-ROM presentations.

Access Time

Random data access time is the typical amount of time required between the request for information and a read operation from CD-ROM. The average access time of most CD-ROM drives is between 100ms (minimum) and 500ms (maximum). This is well below random access time for hard disks (which is normally 15ms), and for RAM (which has an instantaneous random access time). As most interactive multimedia applications shuffle around large amounts of audio and video data during interactive branching, fast access times are important to the flow of the presentation.

Data Transmission Rates

The CD-ROM drive's rotational speed and the density of the "pits" and "lands" on the CD allow data to be read from a single speed disc drive at 150 KBps, a double-speed drive at 300 KBps, a quad-speed drive at 450 KBps, and a 6x speed-drive at 600 KBps. The specified data rate for continuous CD-quality audio is determined by the sector size (of the Red Book[2] sector specification) multiplied by 75 frames/sector. The result yields a data transfer rate of 176 KBps (75 x 2353) for continuous audio data. Furthermore, since the sector size of the Yellow Book specification requires less data (2048) than the Red Book specification (2353), a data transfer rate of 150 KBps is required for continuous audio data from CD-ROM. A quad-speed drive will improve the transfer rate for both audio and video from CD-ROM.

[2] For a discussion on the Red Book and Yellow Book Standards, see Chapter 5.

Internal Buffer Capacity

The performance of a CD-ROM drive also depends on the size of the internal buffer. The internal buffer on a CD-ROM drive consists of a RAM chip that functions as a temporary holding place for data retrieved by the drive. The average internal buffer capacity for a single-speed drive is 16 KB, and 32 KB for a double-speed drive. The internal buffer capacity of a CD-ROM drive has increased to 256 KB with the emergence of quad-speed drives.

The following example illustrates the importance of access time, data rate, and internal buffer capacity.

A sample QuickTime movie (audio and video) of 45 seconds has been digitized and is limited to a data rate of 230 KBps. A double-speed drive with an internal buffer capacity of 256KB and an average data access time of 300ms is provided. Although the drive is capable of a continuous data transfer of 300 KBps, a benchmark rating using a CD-ROM performance utility reveals that it transfers 175 KBps. As a result, the video segment's desired playback rate (230KBps) creates a bottleneck of 55KB (230KB - 175KB). The internal buffer can compensate for approximately 4.5 seconds before there is any noticeable loss of data, if any. In this example the buffer has a 300ms average grace period while the drive accesses the next block of data. The access time and the internal buffer capacity may in fact be sufficient to sustain the data requirements of the video segment. As the access time increases, there is a proportionate decrease in data continuity (reduction of video frames, and interruptions in the audio file). Increasing the internal buffer capacity of a CD-ROM's RAM is essential for digital media components that demand reliable data transmission.

3.6 Data Compression

Data compression promises to be an effective means at alleviating the bottleneck effect inherent with CD-ROM and network transmission medium by reducing redundant data while maintaining to a reasonable degree the original quality. This section will discuss audio and video compression techniques with an emphasis on the MPEG (Motion Pictures Expert Group) compression schemes.

3.6.1 Introduction

First of all it is important to identify the storage capacity required for each media type in terms of bits per second (bps). This is the standard measurement of bandwidth rate for various transmission mediums. The following formulas identify the storage requirements for audio, motion video, graphics, and MIDI in bps.

1) Audio requires (Sampling Rate x bit resolution x 2or1 (stereo or mono)
Ex. (44.1kHz at 16 bit Stereo) = 1.4 Mbps or ~ 10 MBytes per minute

2) Motion Video requires (width x height x color depth x frames per second / compression)
Ex. (320 x 240 x 8 bit color x 15fps) = 9.2 Mbps or ~ 70 MBytes per minute

3) Graphics requires (width x height x color depth)
Ex. 640 x 480 x 8 bit = 2.4 Mbps or ~ 300 KBytes per page.

4) MIDI (Depending on the density of the MIDI file)
Approximately 80 Kbpm. [3]

To transfer this media over a networked system in real-time requires data compression.

3.6.2 Audio Compression

ADPCM (Adaptive Delta Pulse Code Modulation) is the audio compression standard used in CD-i, CD-ROM/XA, and other CD-based gaming formats.[4] It is a lossy (i.e. reduction in quality) compression scheme yielding a maximum of a 4:1 compression ratio. The Macintosh also supports the Macintosh Audio Compression and Expansion (MACE) standard to achieve audio compression ratios from 3:1 to 6:1. Compressing audio with these schemes results in a degradation (dependent on the compression ratio) in the quality of the original file.

[3] This is difficult to determine, as MIDI is a description language.

[4] These formats will be discussed further in Chapter 5.

Perceptual Coding

In recent years there has been a growing demand for audio compression methods that reduce the size of recorded information without sacrificing sound quality. One approach is to utilize perceptual coding schemes. There is considerable debate as to whether compression using perceptual coding schemes is capable of producing high-quality sound reproduction. Some believe the quality to be quite poor while others insist the anomalies are relatively imperceptible. In any case the discussion of data compression using perceptual coding schemes will focus only on the MPEG-1 audio compression standard.

MPEG-1 describes the compression of audio signals using high performance perceptual coding schemes. [18] It specifies a family of three audio coding schemes simply called Layer-I, II and III. All Layers have different complexities and also different target data rates. Perceptual coding is based on the removal of redundant data from an audio signal that presumably cannot be perceived by the human ear.

The redundant data is removed based on a well-established psychoacoustic phenomena. The MPEG-1 Layer I and II encoding schemes analyze the spectral component of an audio signal by filtering the signal into 32 sub-bands. A psychoacoustic model is applied to estimate the just noticeable noise-level for each band in the filterbank. The bits used to encode the signal are further divided among the sub-bands according to the strength of the signal in each band. If the signal is below the threshold of audibility or is masked by another tone, that signal is not encoded.[5]

Table 3.1 identifies the different layers in terms of their target data rates and applications.

Table 3.1

Layer	target rates/channel	applications
I	192Kbps	Philips DCC
II	128Kbps	VideoCD, and Digital Audio Broadcasting (Europe)
III	64Kbps	ISDN

[5] A"masking tone" raises the threshold of audibility of other tones around it.

Reducing the data rate of an audio file (44.1kHz x 16-bit x mono (1 channel) = 704 Kbps) to 64 Kbps (MPEG-1 Layer III's target data rate), requires a compression ratio of 10:1. This reduces the data rate by a factor of 10 while according to many listeners, maintains a high-quality signal.

Although MPEG-1 audio produces a reduction in size and can accurately reproduce the original signal, this format requires additional hardware to decode the audio file once it has been sent to its destination. Real-time decoders remain expensive and are not widely available to the consumer. Until this issue is resolved, MPEG audio in a networked application will remain relatively uncommon. Nevertheless, MPEG audio coding is being used by many commercial enterprises to significantly reduce storage costs and transmission line charges.

3.6.3 Video Compression

MPEG, Cinepak, and Indeo (from Intel) are based on a lossy compression format, and use both interframe and intraframe compression to achieve their target data rate. In a lossy compression scheme, the image quality of a motion video file is reduced as the compression ratio increases. Table 3.2 illustrates this point using a reference digital video file. The target data rate is inversely proportionate to the compression ratio and reduction in quality.

Table 3.2

Video 352 x 240 x 24-bit x 30fps = 7.7MBps

Quality	Target	Ratio
OK	150 KBps	1:52:1
better	300 KBps	1:26:1
best	600 KBps	1:13:1

Interframe compression is compression attained by eliminating interframe redundancy or image segments that remain static between frames. The classic example is the "talking head" shot; where the background remains stable and only face and shoulder movements are encoded. Interframe compression techniques store the background information once, and then retain only the data required to describe the minor changes between the frames.[19] As the amount of movement decreases, the frame rate during data transmission increases. Intraframe compression is attained by eliminating redundant data within a frame, without reference to successive frames.

MPEG-! was designed to provide VHS quality digital video at a data rate of 150 KBps. MPEG-1 is based around the Standard Image Format (SIF) of 352x240 that reduces the equivalent of 7.7 MBps full bandwidth video (352 x 240 x 24-bit x 30) to 150 KBps.

A compression ratio of 52:1 was needed to support MPEG-1 video for CD-ROM. In comparison, the Cinepak and Indeo software compression schemes offer only a maximum compression ratio of 5:1 for motion video.

3.7 Discussion

The technologies discussed in this chapter are constantly improving. Real-time communication protocols (i.e. UDP), and data compression (i.e. MPEG) allow media to be transmitted over the Internet in real-time with fairly reliable results. The HTTP protocol allows hypermedia documents to be exchanged between a large audience and interactive applications to be downloaded into a browser over the Web. Although the impact of what these technologies may offer networked music systems is not fully known, there have been a growing number of innovative applications being implemented for the Web and CD-ROM. These applications will be discussed further in Chapter 4.

Chapter Four
Networked Multimedia Applications

4.1 Introduction

Internet applications began with FTP, electronic mail, and telnet. Later, text-based virtual reality environments based on Multi-User Dungeons (MUD), and MUD Object-Oriented environments (MOO) emerged. [1] The Internet now supports hypermedia documents, networked music applications, audio-on-demand, and videoconferencing.

[1] MOOs are based on the LambdaMOO server initiated by Curtis Pavel at Xerox Parc in 1992. LambdaMOO is a network-accessible, multi-user, programmable, interactive system well-suited to the construction of text-based adventure games, conferencing systems, and other collaborative software. Its most common use, however, is a multi-participant, low bandwidth virtual reality. From the LambdaMOO manual on the World Wide Web.

4.2 Networked Hypermedia Browsers and Applications

With the emergence of Web browsers such as Mosaic, Netscape and HotJava, the World Wide Web has been bombarded with a myriad of new applications promising to provide interactive multimedia experiences.[20]

4.2.1 Netscape: Hypermedia Document Browser

Netscape Navigator™ provides access to an almost infinite number of digital hypermedia documents over the Internet. It is best suited for applications that do not require media-on-demand and real-time interaction between client and server.

In the pre-release specification of Netscape 2.0, the notion of "browser" has been replaced by the idea of an open environment for the development and distribution of interactive multimedia content. Netscape now supports concurrent streaming of digital audio and video (using RealAudio), in-line applications (QuickTime, Director movies, Java applets), and a scripting language used to integrate these features into an HTML document and for interactive control over various media components.

As Netscape continues to shift its focus towards interactive multimedia content, the problem of excessive download delays will become even more prominent. It is still unclear whether Director presentations will need to be downloaded with the Web page, or if they will be streamed over the Internet from a Web server. In any case, the temporary solution to the bandwidth problem is to access the media locally on CD-ROM. [2]

The growing number of Web-MOOs, Interactive Web-Games, collaborative Web-Art, and Web-Music spaces, may be attributed to new media artists exploring the potential of the Web as a future stage, digital playground, or virtual space for their creations. Until the infrastructure of the Web supports media-on-demand, long delay times will continue to reduce the appeal of these types of works.

[2] This issue will be discussed further in Chapter 8.

4.2.2 HotJava: Interactive Presentations on the Web

The HotJava browser from Sun Microsystems is used to view and interactive with Java applets.[21] Java applets are implemented using the Java programming language created by Sun and are used to make interactive multimedia presentations for the Web.

Applets need to be downloaded before they can be used within a Web browser window. Several interactive Java music applications have been created such as *CatBand*. [22] *CatBand* is an excellent early example of Java's capabilities. This applet permits the end-user to collaborate with the on-screen cats using the mouse and computer keyboard to trigger sound events and computer animation.

4.2.3 VRML: 3-D view of the Web

Three-dimensional visualization tools integrated within the Web allow one to view and retrieve data with a different perspective than that commonly available with HTML. The Virtual Reality Modeling Language (VRML) is used to describe multi-participant interactive three-dimensional environments or "virtual worlds." VRML emerged in 1994, at the first annual World Wide Web Conference in Geneva, Switzerland. It was proposed as a common language and API (analog of HTML for virtual reality) for specifying 3-D scene description and Web hyperlinks.[23][24] Web pages which integrate VRML can contain objects that refer to other environments. HTML documents or other valid MIME types. VRML's API is based on the Open Inventor ASCII File Format from Silicon Graphics, Inc., with extensions to support networking. The Inventor File Format supports complete descriptions of 3-D scenes with polygonally rendered objects, lighting, materials, ambient properties and realism effects. A VRML "viewer" is required to view or interact with VRML documents within Netscape.

While HTML was never intended to be used as a scripting language to implement interactive music spaces, many artists have embraced HTML as a means to distribute new media works. VRML offers far more in this respect than HTML. It is for this reason that the Web will most likely emerge as a medium to support innovative music environments.

There has also been some discussion about implementing a VRML extension for sound spatialization. Graphically-defined objects could be used to trigger sonic events on the basis of user input. [25]

4.3 Real-Time Networked Digital Media

While the following real-time applications were designed for other applications than music, their compression algorithms and real-time communication protocols could be important in developing networked interactive music systems.

4.3.1 RealAudio

RealAudio is a networked application that provides audio-on-demand to Internet users. [26] It is based on a proprietary compression algorithm and real-time communication protocols (UDP) that allow voice to be transmitted over the Internet using a 14.4 Kbps modem and music using a 28.8 Kbps modem. The RealAudio delivery and playback system includes the RealAudio Encoder (encodes audio files in the RealAudio format), RealAudio Server (distributes broad-based multicasting audio streams to connected users) and the RealAudio Player (allows connected users to access RealAudio programming for real-time playback) (See Figure 4.1)

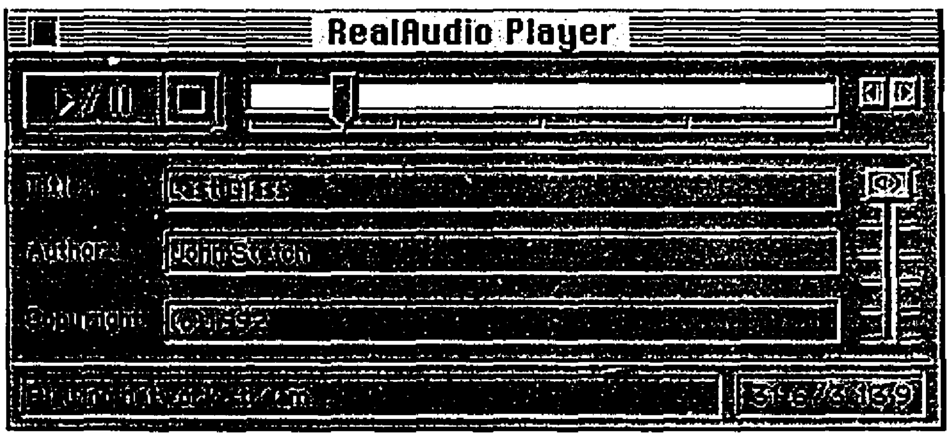

Figure 4.1: Illustrates the RealAudio Player and its interactive features.

RealAudio's compression algorithm reduces data rates from 176 KBps (digital audio file) down 3.5 KBps (music compression). This compression ratio (50:1) is essential in order for music to be transmitted over a 28.8 Kbps modem. While the RealAudio system ensures real-time audio over a network structure, it provides no guarantee that all data packets will arrive at their destination. Figure 4.2 illustrates the percentage of packets that are lost during a sample data transmission.

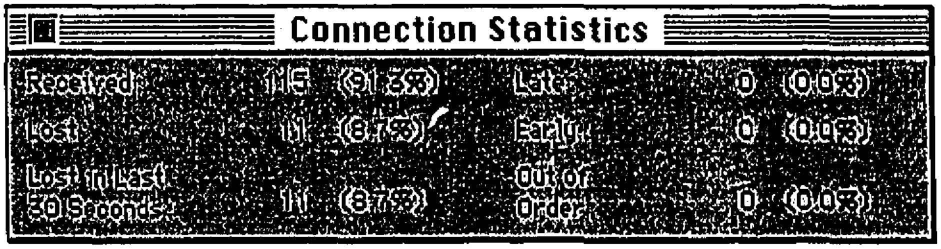

Figure 4.2: Illustrates the connection statistics between client and server.

The buffering and caching system ensures a data rate of 1 KB or 3 KB (depending on the type of audio) of data per second. This technology is needed for non-interrupted audio streams over narrowband devices. (See Figure 4.3).

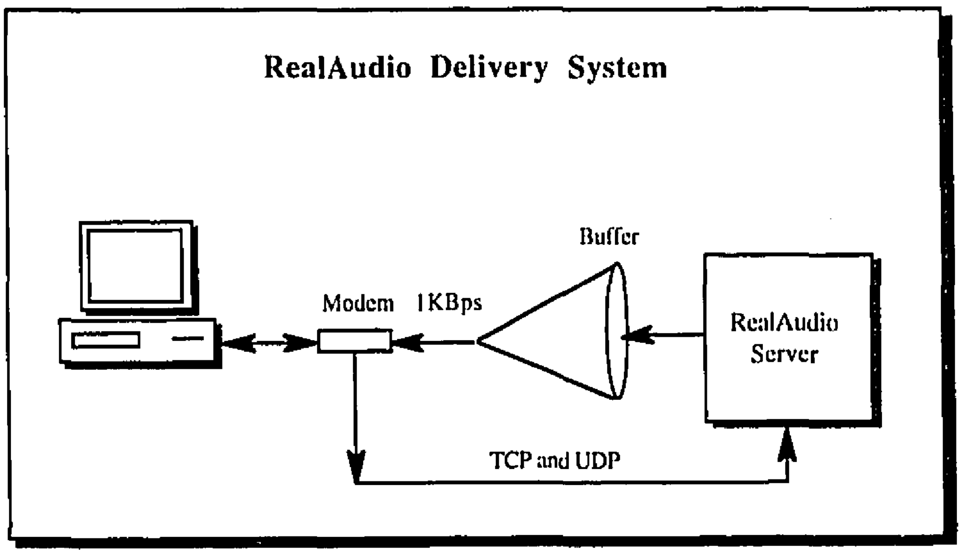

Figure 4.3: Basic design of the RealAudio delivery system.

Future Trends and Technology

The promise of sending audio streams in real-time across the Internet appeals to those who cannot tolerate the long download delays inherent with FTP and Web audio.

The RealAudio system allows independent recording artists, sound designers, composers, symphony orchestras, and even "garage bands" to broadcast their concerts in real-time to an Internet audience. Progressive Networks has thus reinvented an old medium (Radio) over a relatively new medium (Internet) by means of a new technology (RealAudio). For example, the Canadian Electroacoustic Community (CEC) is broadcasting a series of concerts using RealAudio over the Internet from Dec. 17 to 31, 1995.

4.3.2 CU-SeeMe

CU-SeeMe is a low-cost videoconferencing tool that allows Macintosh end-users with the proper hardware (digital camera, and a connection to the Internet) to send and receive audio-video streams in real-time. [27] Digital media streams are sent out as unicast IP (point-to-point) or as multicast IP (point-to-many) packets over the Internet. In the unicast mode, participants directly connect to the IP address of another CU-SeeMe participant. This allows participants to communicate digital media between one another over a privatized "virtual channel." As Macintosh computers do not support multicast IP, those users who wish to participate in a multicast session (broadcasting video and audio streams to multiple participants), need to connect to "reflectors" (UNIX machines running multicast software). A maximum of sixteen participants can be connected to a reflector at a time.

A CU-SeeMe video image has a 160 x 120 pixel resolution, 4-bit grayscale, with a maximum of 4 frames per second (fps). A constant data transmission rate of 9 KBps (160 x 120 x 4 bit x 1 fps) is needed to support motion video, and 176 KBps for digital audio. Having audio and video as the primary media in a real-time application clearly requires fast data transmission lines and data compression. CU-SeeMe compresses the audio and video as it is being digitized.[3] The compression ratio per media, is determined by an adjustable data rate that is influenced by the bandwidth capabilities of the transmission medium (14.4 Kbps modem). If that rate exceeds the capacity of the transmission medium, data will be lost. This is evident in Figure 4.4, where 50% percent of all data transmitted has not been received.

---

[3] Only transmits the portions of a picture that have changed significantly from the preceding frame.

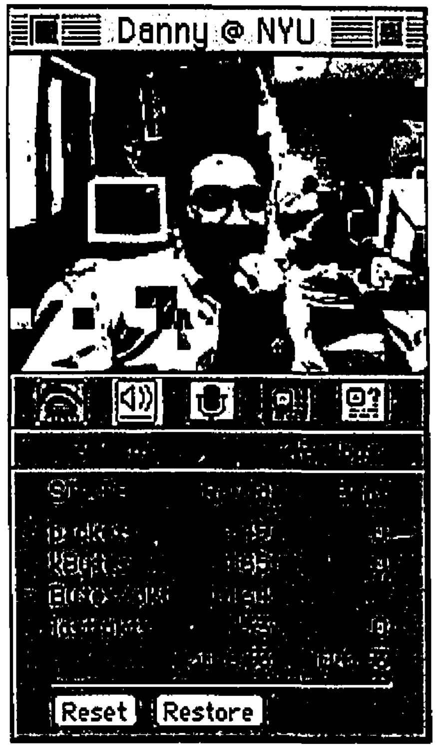

Figure 4.4: A typical CU-SeeMe session illustrating that 50% of the packets have been lost. This would be unacceptable in a networked music environment that is dependent on accurate data.

CU-SeeMe provides the infrastructure for a personalized broadcast medium to ambitious new media artists and composers. *The Slowest Train in The World* on June 2nd 1995, was a recent concert featuring CU-SeeMe technology. This concert contained a program of live computer music from various ICMA members. [28]

While high-fidelity digital audio is an essential component for rendering these compositions effective, the novelty of CU-SeeMe has blurred its actual capabilities. It is evident in participating in a CU-SeeMe session, that the technology cannot support continuous and reliable transmission of musical and visual information. As a result, its use as a virtual concert hall or performance tool at the moment seems premature.

4.3.3 Applications of the MBone

The MBone or Multicast Backbone is a virtual network layered on 'top' of the Internet. It originated in 1992 from an effort to multicast audio and video from meetings of the Internet Engineering Task Force (IETF). [29]

Unlike standard Internet communication methods (unicast IP—one-to-one), participants or hosts on the MBone communicate with one another using IP multicast. The Mbone applications such as *vat* (Audio Conference Tool), *Nevot* (Audio Conference Tool), *NetVideo* (Video Conference Tool), *nv* (Video Conference Tool), and *wb* (Shared WhiteBoard Tool) are all based on IP multicast.[30]

Since the Internet only supports unicast IP, multicast IP packets are encapsulated within regular unicast IP packets that pass through virtual 'tunnels' from an 'mrouter' (routing deamon that supports multicast IP) to conventional Internet IP routers. As the packet reaches its destination, the unicast IP header is removed, and the packets return to multicast form. The multicast IP routers take the responsibility of distributing and replicating the multicast data stream to participants with multicast support. The MBone is limited to computers which support IP multicasting. (See Figure 4.5)

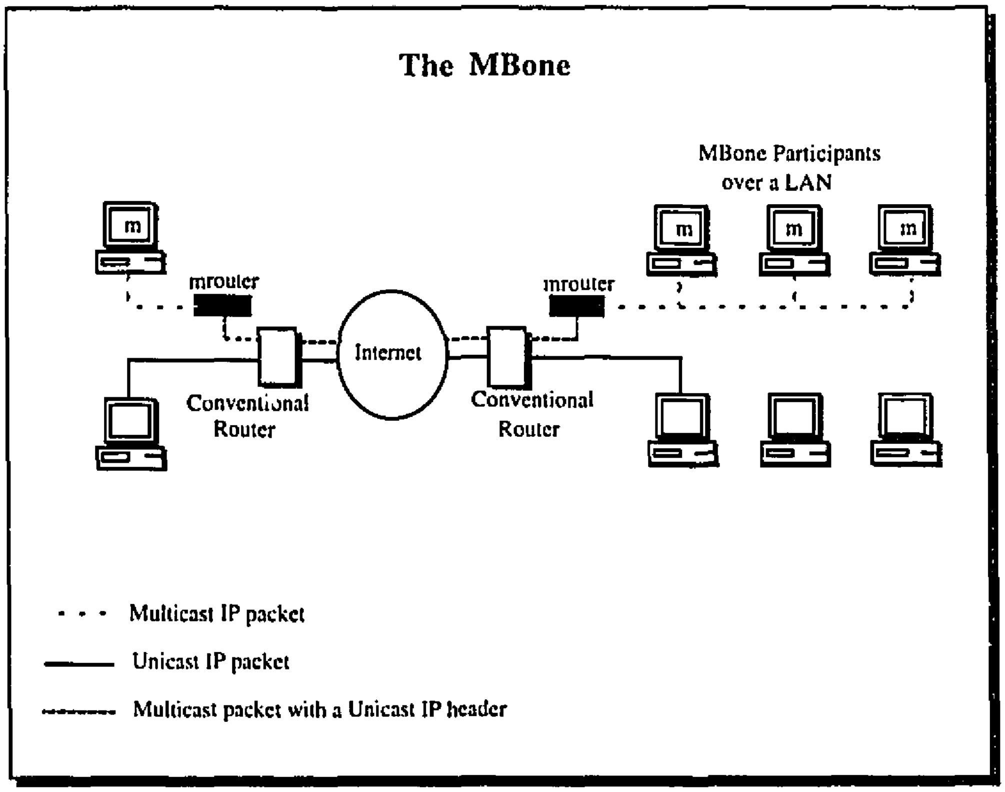

Figure 4.5: Basic architecture of the MBone virtual network.

The transmission capacity of the MBone network has been initially set to a maximum of 500 Kbps for the entire MBone community. Approximately a quarter (128 Kbps) of that bandwidth is reserved for video at 4-5 fps (frames per second). The remaining bandwidth is used for audio channels as well as multi-user whiteboard data. The number of parallel MBone sessions that can be transmitted or broadcasted is limited to the maximum data transmission of 500 Kbps. There is limited space for concurrent video and/or audio sessions, therefore it is imperative that new media artists be aware of these limitations in their broadcasts. With the maximum bandwidth of 500 Kbps up to seven simultaneous audio channels are possible at 64 Kbps per channel if no video is transmitted. Although the audio is of telephone-quality and the video capabilities limited to 128 Kbps, the ability to transmit and receive video and audio in real-time is the first instance of true videoconferencing capabilities on the Internet. Despite these limitations the MBone is being used to multicast live rock concerts, [4] computer music concerts, and radio broadcasts. [5]

The MBone has succeeded in creating a dynamic multimedia collaborative environment over the Internet. It is a prototypical environment for artists who would like to distribute their digital media to a vast audience in real-time. At this time, however, the limited bandwidth and the need for sophisticated computer hardware and software make the MBone and its applications impractical for most commercial purposes.

4.4 Networked Music Environments

The fact that networked music applications have not been fully explored may be attributed to unreliable communication protocols and data transmission mediums. The following systems provide the architecture for interactive networked systems based on MPEG-1 audio and MIDI.

4.4.1 MIDI and Audio over ISDN

ISDN is capable of transmitting real-time digital audio using various data reduction algorithms based on perceptual coding methods. MPEG-1 Layer III provides the best

---

[4] The Rolling Stones multicasted 20 minutes of their November 18, 1994 Dallas Cotton Bowl Concert as a promotion for a subsequent pay-per-view TV special. Weiss, Aaron. "Stretching the MBone: The Internet Broadcasting Network." *Internet World* March 1995: 38-41.

[5] Radio broadcasts, perhaps because of their lesser bandwidth requirements, have become common on the MBone. Examples include Big Byte program on BBC Radio 5, which offered users a feedback opportunity via an MBone whiteboard and IRC (Internet Relay Chat) connection. Weiss, Aaron. "Stretching the MBone: The Internet Broadcasting Network." *Internet World* March 1995: 38-41.

audio quality for "low" bit rates around 64 Kbps per audio channel. Transmitting MIDI data over ISDN requires less bandwidth and is also more efficient in that it transmits musical gestures and performance controls that can easily be altered to suit an interactive performance situation.

A recent study demonstrated a system for transmitting MIDI data and MPEG-1 Layer III audio over ISDN.[31] This system can be categorized into five stages: 1) transmit MIDI data over ISDN; 2) map MIDI data to a preset stored on a sound module; 3) encode the audio from the sound module into MPEG-1 Layer III; 4) transmit the compressed audio to the client over ISDN; and 5) decode the audio stream and play it back through a sound system. In this example, MIDI played on one machine was heard at another and vice-versa for the audio signals.

The client is equipped with a MIDI controller and an MPEG decoder. The server encodes the audio as MPEG and transmits that data over ISDN to the client's speakers. As shown in Figure 2.15, a UART microprocessor and a parallel-serial shift-register is synchronized to the ISDN network clock and is used to convert the asynchronous MIDI transmission rate of 31.25 Kbps to the ISDN 64Kbps synchronous data transmission rate. (See Figure 4.6)

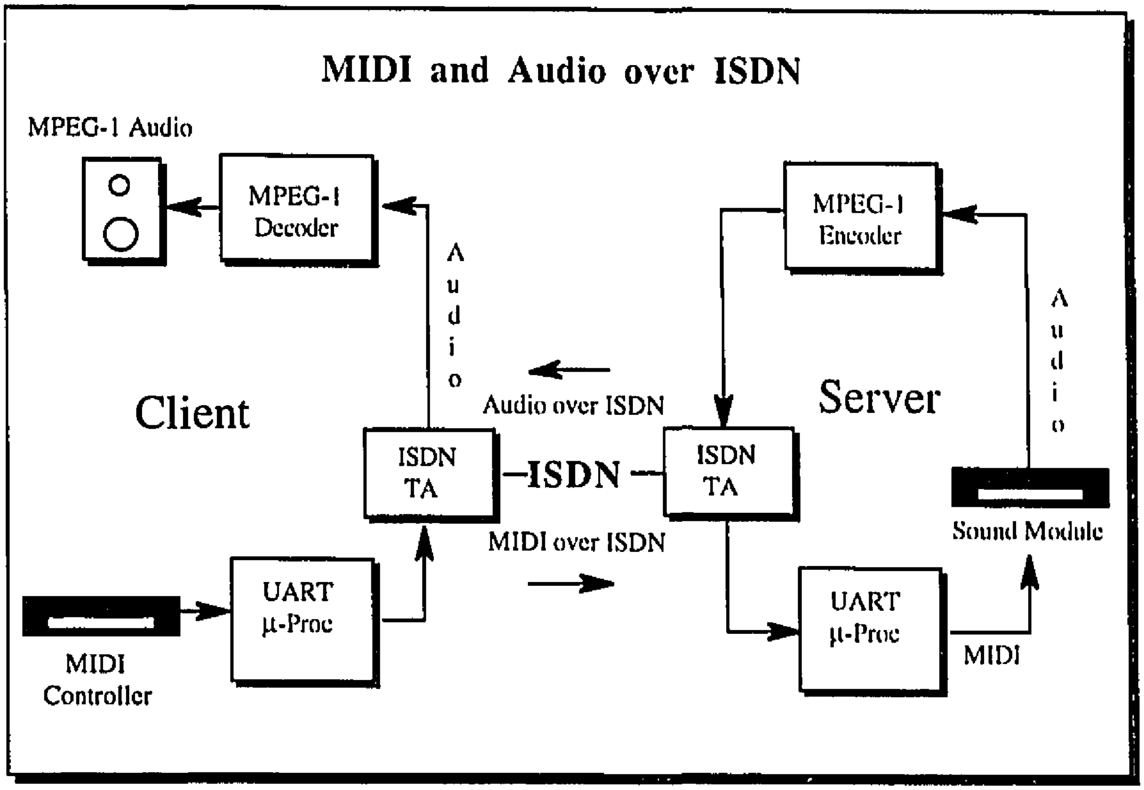

Figure 4.6: Illustrates the MIDI and Audio over ISDN system described in the work of Ole Nielsen

Other Approaches

Other approaches using the ISDN system described above could include remote recording sessions and remote interactive performance environments.

It is possible to imagine a remote recording studio that receives musical information (MIDI or MPEG-1 Layer III audio) over an ISDN line from remote musicians. Artists could send their performances over the ISDN B-Channels (64 Kbps), and communicate back and forth with a studio over the ISDN D-Channel (16 Kbps).

In some cases, computer music compositions require an elaborate setup of sound modules, digital signal processing units, and advanced computing technology. These pieces often require the composer be present in the concert hall. Some compositions that require such a complex setup are often not performed. Using ISDN, an artist could "dial-up" the composer's remote computer and automated system, and perform a composition in a remote concert hall setting. As the MIDI or audio data is transmitting over ISDN, it could then be further processed in real-time on the composer's "personal music machine." This would certainly facilitate both composer and performer demands, and would probably ensure more performances of the work.

4.4.2 Networked Animation Controlled by MIDI

A networked animation environment controlled by MIDI messages could be implemented using Max as a music generator, Director as the virtual stage for cast members to animate, and MIDIShare as the communication protocol between Max and Director over an Ethernet system. (See Figure 4.7)

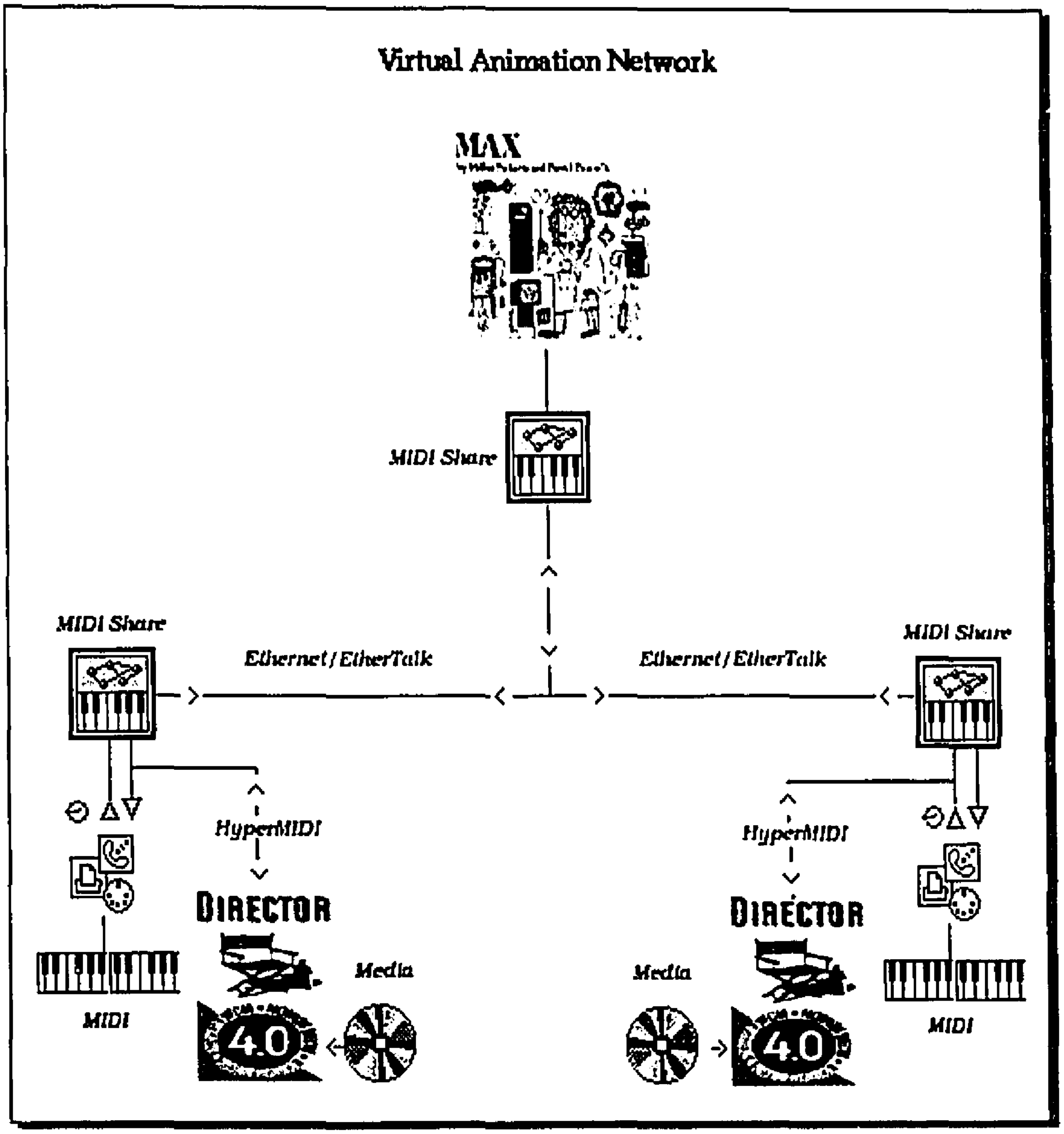

Figure 4.7: Illustrates a possible architecture for a real-time animation system controlled by MIDI data.

Max is an interactive graphic object-oriented programming environment for interactive composition and performance systems. Max was developed at IRCAM by Miller Puckette and David Zicarelli, and was subsequently released in 1990 by Opcode Systems, Inc. Max is based on an object-oriented discipline of self-contained processing units (objects) that communicate with other objects by passing messages via "patch cords." (See Figure 4.8)

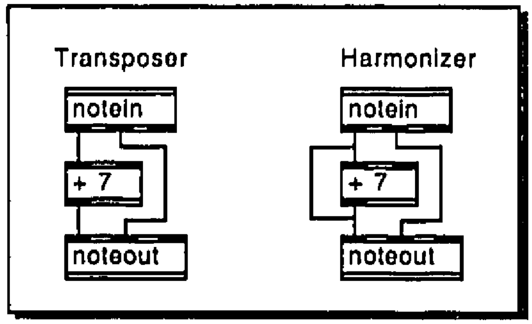

Figure 4.8: A simple Max program for transposing and harmonizing incoming MIDI note values.

MIDIShare is a real-time multi-tasking MIDI operating system developed specifically for musical applications. It contains a suite of tools that allow MIDI applications to exchange data between one another over Ethernet. It is based on a client/server model that is composed of several interconnected component managers for reliable data transmission of musical information over Ethernet.[32]

As shown in Figure 4.9, MIDIShares msPass application (2 inputs and outputs) sends MIDI data to HyperMIDI from a remote computer. This data is used to trigger cast members within Director.

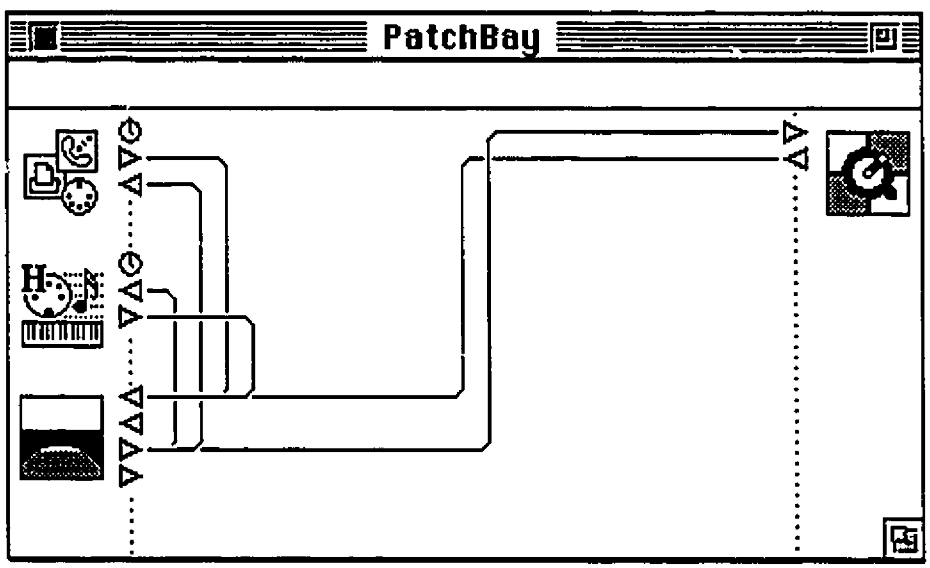

Figure 4.9: Illustrates a virtual connection

A Max patch may be used to generate music or control data to animate cast members within Director. As shown in Figure 4.10, Max sends MIDI data to the msPass utility.

This data is then sent over Ethernet to the msPass utility residing on the clients computer.

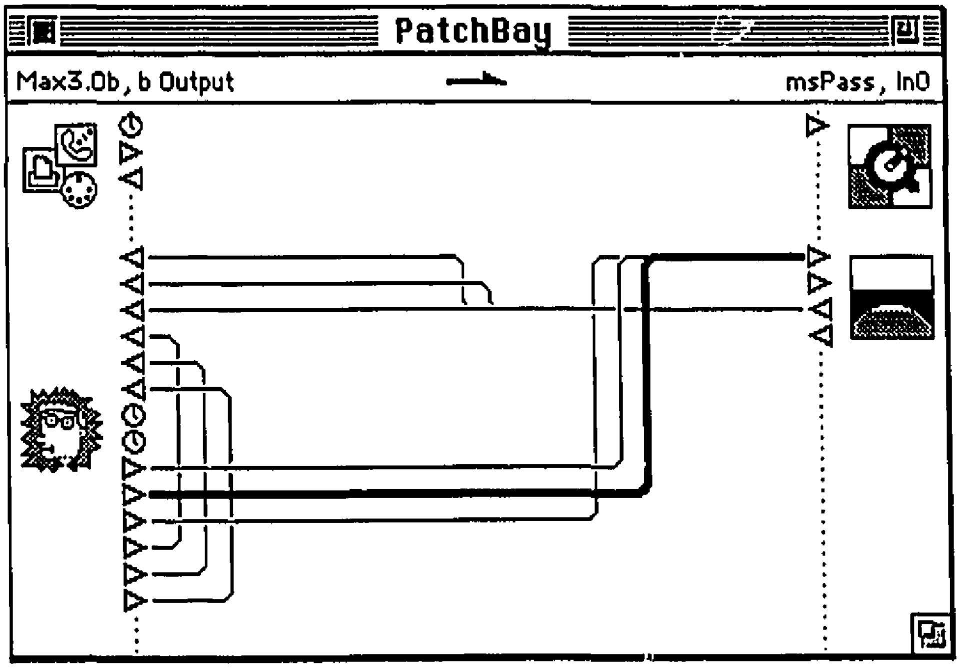

Figure 4.10: Illustrates how Max receives and sends its data to msPass using Apple's PatchBay.

Max communicates with Director through HyperMIDI commands such as:

```
hmOpenMIDI "8000 8000 8000 ",8000 // create 3 input ports, each with an 8K buffer
put hmMIDIfile("read",it,"msec") into theSeq  // read MIDI data and store it into theSeq var.
hmWriteMIDI theSeq    // start playing.
```

Appendix 4.1 provides more of Director's Lingo code examples and HyperMIDI XCMDs to demonstrate how MIDI data could be used to control the movement and behaviours of cast members on-screen.

Discussion 4.5

Other than the work shown in section 4.4.1 (MIDI and Audio over ISDN) and 4.4.2 (Networked Animation), networked music systems have not been fully explored. The technologies used in RealAudio, CU-SeeMe, Netscape, and the MBone are more significant for networked music than the programs themselves. Networked music systems would require that a Web browser support compiled in-line Max applications. These applications could appear within a hypermedia document to permit the end-user to interact with a Max patch over the Web. This would be similar to interacting with Java applets and Director movies. At the moment Max contains the tools needed to create interactive music systems, but not over a network.

Chapter Five
Compact Disc Presentation Schemes

5.1 Introduction

Compact disc (CD) technology has sparked a major creative and commercial revolution not only in the recording industry but in the desktop and set-top CD-based entertainment industries as well. An examination of the various presentation schemes for desktop and set-top systems will demonstrate the ramifications of CD technology.

The most prevalent CD format or presentation scheme is the CD-DA or audio CD as described in Sony and Philips Red Book Standard specification.[33] Further developments in 1985, resulted in the CD-ROM format as described in the Yellow Book Standard specification. The impetus for the CD-ROM was to encode computer data (audio, motion video, graphics, and images) and digital audio on one CD.

The vast array of presentation schemes based on the Red Book standard may be attributed to the realization by "new media" developers, recording artists, interactive media industries, and publishers that their multimedia data could be stored, played, and delivered in the digital domain. Due to the wide variety of CD-based applications, (electronic reference books, games, digital photographs, digital video, and interactive music CDs) there are new formats surfacing every year. All subsequent multimedia CD formats are a subset of the Yellow Book standard. These presentation schemes generally fall under two categories: desktop and set-top CDs.

5.2 Desktop Presentations Schemes

Desktop CDs are computer-based formats that are playable on CD-ROM drives connected to Macintosh, MPC, and UNIX computers. The encoding schemes, and the software drivers used to recognize and process the data are based on the different system architectures of these platforms This is a fine detail and is only important when mastering the CD. Desktop CD formats fall into four categories: Yellow Book, interleaved, mixed-mode, and multi-session.

5.2.1 CD-ROM: The Yellow Book Specification

There are two main categories of the Yellow Book Standard: International Standards Organization (ISO) 9660 and the Mac Hierarchical File System (HFS). The ISO 9660 format was the first attempt to create a CD accessible to CD-ROM systems. It has the capacity to store approximately 550 MB and is the standard format for PC Windows, PC DOS, and Unix CD-ROM systems. The Mac HFS can only be read on a Macintosh computer system and has a maximum storage capacity of 650 MB.[1]

CD-ROM was the first storage format to offer the possibility of non-linear interactive presentations.[2] The first non-audio CD products were encyclopedias and reference works. Since then, CD-ROM has become the primary delivery medium for interactive

[1] The Macintosh HFS format has a larger storage capacity because it does not require the same amount of error correction as the ISO 9660 format.

[2] The audio and computer data (i.e., audio, digital video, graphics, text, and multimedia application) are separate components on the CD-ROM and are accessed through the multimedia applications scripts.

music (e.g. *Xplora1: Peter Gabriel's Secret World*), interactive role-playing fiction (e.g. *Myst*), and electronic reference books (e.g. *Encarta*, and *The New Grollier Multimedia Encyclopedia*). There have been other innovations based on the CD-ROM format including: interleaved, mixed-mode, and multi-session capabilities.

5.2.2 Interleaved

Prior to interleaved formats only a single media component could be accessed from CD-ROM. [3] As the computer data is interleaved with the digital audio these formats allowed for multiple media to be presented concurrently. In 1988, a joint project between Warner New Media, JVC and Sony resulted in the development of the CD+G (CD plus graphics), CD+M (CD plus MIDI), and CD-ROM/XA (eXtended architecture) presentation schemes.

The duration time of Beethoven's *9th Symphony* determined the 74-minute playing time of a digital audio recording on one audio CD, yet most popular recordings do not exceed 50 minutes. This leaves approximately 25 minutes for non-audio data. As shown in Figure 5.2 there are 6 bits of subcode information left undefined in the Red Book specification, resulting in an extra 25MB can be used for computer data.

Red Book CD frame format

Sync 24 bits	Subcode 8 bits	User Data 96 bits	Parity 32 bits	User Data 96 bits	Parity 32 bits

Figure 5.1: Identifies the Red Book CD frame format. There are 6 bits of the 8-bit subcode region that were undefined during the specification. As a result these extra 6 bits are being used for computer data

Interleaved formats use the undefined 6 bits of the subcode region of an audio CD for multimedia data (i.e., graphics, text, and MIDI). The multimedia content is synchronized to the recording , and is often an accompaniment to the audio presentation, such as lyrics for karaoke, still images, and biographics.

[3] This is the limitation of the laser beam used in the CD-ROM drive. It does not permit multiple places on the CD to be accessed at once, and as a result, only one component can be retrieved at time.

CD+Graphics

The quality of the multimedia content is limited by the 25MB storage capacity reserved for the multimedia data. Approximately 850 still images at 4 bit color (16 colors) and 288 x 192 pixel resolution can be stored. The data rate of all subcodes in the Red Book specification is approximately 58 Kbps as shown in Figure 5.2. For interleaved formats the data rate for the graphic channels is reduced to 44 Kbps.[4] The 8 subcode bits are collected from 98 consecutive frames to form a subcode data block. The data block playback rate from CD-ROM is 75 blocks per second. Each block consists of 32 frames at 73.5 bytes per frame resulting in sector size of 2352 bytes. [34]

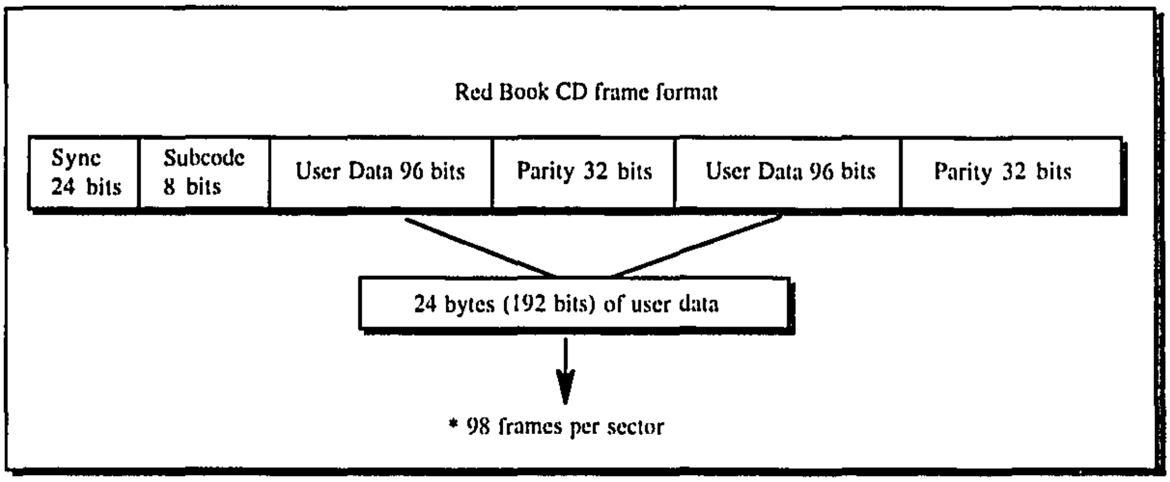

Figure 5.2: The data rate for subcode regions is determined by 75 (blocks/sec) x 8 (data channels) x 98 bits (per channel) = 58 Kbps.

The data rate of 44Kbps per graphics channel is not sufficient for synchronous full-frame digital video and audio.

Originally, there were only two systems (NEC's TurboGraphix-16, and JVCs CD-audio + graphics player) that could support CD+G discs. Audio CD players were not equipped to decode computer data and as a result the visual element to these presentations could not be viewed. In 1988, *Gustav Holst: The Planets*, was the first

[4] The data rate for multimedia content from CD-ROM is calculated as 75 (blocks/sec) x 6 (data channels) x 98 bits (per channel) = 44 Kbps.

CD+G title to be released. Since that time, approximately 200 CD+G titles have been developed.

CD+MIDI

The CD+M presentation scheme encodes MIDI data in the same location (subcode region of an audio frame) that the CD+G format encoded its graphics and other data.

The impetus for CD+M was to provide artists with the ability to encode additional musical information (MIDI) interleaved with the digital audio recording. Since the MIDI data is synchronized to the audio data on the CD, it is possible for the listeners to re-orchestrate the music. The MIDI specification provides 16 MIDI channels and it is therefore possible for an artist to supply 16 additional MIDI tracks directed to a 16 voice multi-timbral MIDI keyboard to accompany a solo flute recording. The potential for this format was almost limitless; however, it would have required peripheral devices and software to create interactive musical environments.

The only player which supported the CD+M format, was JVC's XL-G512. The player featured MIDI and video output jacks, yet it did not have sound capabilities—requiring an external sound module to hear the additional tracks. If General MIDI sound samples had been built into the player it could have been commercially successful; as no peripheral devices would have been required. There are very few CD+M titles, one of which is Warner New Media's *The Magic Flute* based on Mozart's Opera.

Although CD+G and CD+M appeared to be innovative presentation schemes, they were more suited toward a niche market and were perhaps too esoteric for the average consumer.

CD-ROM/XA

CD-ROM extended architecture (XA) is essentially an extension of the Yellow Book CD-ROM format. It is very similar in concept to both CD+G and CD+M formats for interleaving data. The only difference is that it can encode two levels (Level B and C) of digital audio. Level B allows four 4-bit stereo or 8 concurrent mono tracks at 37.8 kHz to be recorded. In contrast, Level C allows 8 stereo or 16 concurrent mono tracks at

18.9 kHz and 4-bit resolution to be recorded. Audio recording is based on the Adaptive Delta Pulse Code Modulation (ADPCM) compression scheme to allow more audio to be encoded on the CD-ROM, and also to provide more concurrent audio tracks.

CD-ROM/XA offered very little in terms interactivity and random access, yet it was used as the presentation scheme for many of the first classical music CD-ROMs.

5.2.3 Mixed Mode

In 1991, CD+ G, CD+M, and CD-ROM/XA were the most prevalant new consumer formats for music-based multimedia. The recording industry adopted these formats predicting that all popular music recordings would be enhanced with multimedia features by 1993.[35] Although these formats were significant innovations, this prediction did not happen and these formats have been long forgotten.

The mixed-mode CD format was the next format to be introduced. Unlike interleaved formats, data encoded on a mixed-mode CD occupies it's own disc partition. This allows media to be randomly accessed without being associated or referenced to a particular audio track. Thus computer data and Red Book audio may be partitioned in separate locations on a CD. According to the mixed-mode CD specification, computer data was to be encoded in the first track while subsequent tracks were used to encode digital audio. CD-ROM drives and standard audio CD players could decode mixed-mode data, but problems occurred when the audio CD player would attempt to play the first audio track. [5]

Mixed-mode CDs have had reasonable success, especially in products that integrate interactive front-ends to the linear audio.[6] However, it was not an effective marketing scheme to expect consumers to skip to "Track 2" of an audio CD for each listen, and it therefore failed.

[5] Record companies were reluctant to produce titles for the more traditional CD-ROM format because mixed-mode CD-ROMs would not play easily on conventional audio CD players. Placing a CD-ROM in an audio player will either skip track one —because CD-ROMs use the first track of the disc to store computer readable data—, blow out ones speakers, or not play at all. Most CD-ROMs only have one track. Mixed mode CD-ROMs use the first track for computer data and the remaining tracks for music.

[6] U2 Zooropa mixed-mode CD is an example.

5.2.4 Multi-session

When CDs were first introduced in the early-80s and CD-ROM in 1985, there were no applications that required multiple sessions or the need to add or extend data onto a CD after it had been pressed. As more diverse applications emerged, such as Kodak's Photo CD format, there was a need to add data (photo graphics) to a disc at a later time; hence multi-session. This development also proved to be the solution to the "Track-1" problem inherent with the mixed-mode CD format.

All CD presentation schemes use a "lead-in" section of the disc to store the *table of contents* of the actual data recording onto the CD and a "lead-out" area defining the end of that session. As illustrated in Figure 5.3, however, a multi-session CD consists of a number of such sessions each with its own lead-in, data, and lead-out area. The CD player requires the table of contents from the lead-in area to position itself at the right location to play that particular session on the CD.

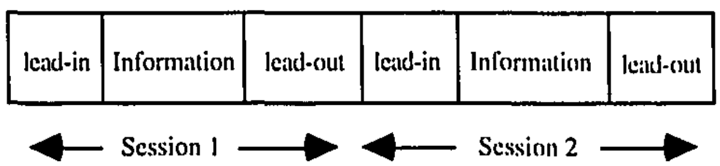

Figure 5.3: Illustrates a multi-session format.

One such use of the multi-session format for music is the Enhanced CD. The Enhanced CD format uses the multi-session presentation scheme to store linear audio in the first session and computer data in subsequent sessions of a multi-session CD.[36] Most audio-CD players were not designed to expect anything other than single session recordings. For that reason, anything past the first session is completely ignored by the player. Conversely, CD-ROM drives can play back the audio session and subsequent sessions of interactive multimedia content.

5.3 Set-top Presentations Schemes

Set-top interactive CD formats are designed to play on consumer gaming platforms connected to a standard television set. As the set-top systems have become a billion dollar industry, it has become imperative that *new media* artists and developers consider the set-top world as a possible distribution medium for their work.

The Set-Top CD formats are subsets of the Yellow Book standard, yet each encode proprietary information on the disc to prevent one manufacturers CDs from playing on another's platform. There are three categories to set-top CD presentation schemes: 1) proprietary gaming CD formats including CD-i (Compact Disc-interactive), 3DO, SegaCD, Sony PlayStation; 2) Video CD; and 3) High Density Compact Disc (HDCD) and Digital Video Disc (DVD).

5.3.1 Proprietary Formats

There has been a wide selection of multiplayer gaming platforms, each claiming to out-perform its successors. The hardware for these platforms support (with add-ons) universal standards such as audio, Photo, and Video CDs.

The Green Book standard was introduced in 1988 as an extended architecture and operating system for Philips/Sony proprietary CD-i discs and players. The CD-i format provides for various levels of audio quality (ADPCM compressed audio) and storage. (See Table 5.1)

Audio Level	Sampling Rate	Resolution	Concurrent Channels	Storage in stereo
* CD-audio	44.1 kHz	16-bit	2 stereo	10.5 MBpm
Level A	37.8 kHz	8-bit	2 stereo or 4 mono	4.53 MBpm
Level B	37.8 kHz	4-bit	4 stereo or 8 mono	2.26 MBpm
Level C	18.9 kHz	4-bit	8 stereo or 16 mono	1.13 MBpm

Table 5.1: *Although CD-i format does not directly support Red Book audio it supports six types of digital audio formats—three quality levels in either mono or stereo.

An extension to the CD-i format is the CD-i "ready disc"— which is essentially a Red Book audio disc with computer data stored after track-1 in an area called the *pregap.*

In 1993, 3DO introduced a new interactive gaming platform. The 3DO CD format is similar to the encoding scheme used for the CD-i format except that the 3DO format allows for CD-quality audio to be encoded. SegaCD and other set-top presentation schemes are based on similar encoding schemes.

5.3.2 Video CD

The Philips Video CD format is a partial solution to Hollywood's desire to offer consumers full- length feature films on one CD. Video CD uses the MPEG-1 video compression format for set-top CD players as described in Philips White Book Standard specification.

Video CD is the White Book standard describing an MPEG-1 linear video format for set-top CD players.

The Video CD format has a storage capacity of 74 minutes of MPEG-1 quality video. The image resolution of MPEG-1 video is frequently promoted as equivalent to VHS VCR video quality image. [37] There has been a wide-spread deployment of MPEG-1 hardware for set-top gaming platforms and desktop MPCs and Macintosh computers. Yet, according to Phil Dodds of the Interactive Multimedia Association (IMA), major movie studios are bypassing MPEG-1 (Video CD) in anticipation of MPEG-2.[38] The Video CD format is significant, however, in that it has furthered the development of improved data transmission rates and storage capacity for CDs.

5.3.3 High Density CDs

Time Warner/Toshiba DVD and the Sony/Philips HDCD format specifications are focused on providing full length digital video movies on compact disc. The specification for both formats is based on the MPEG-2 digital video compression format, and is vying to replace the Video CD. A data transmission rate of approximately 5.6 Mbps is required to support continuous MPEG-2 video streams; a significant increase over MPEG-1. As the data rate increases, there is a proportionate decrease in the amount of digital video that can be encoded on the disc. The use of MPEG-2 video will therefore require an increase in both data transmission rates and storage capacity of a CD.

Although these formats are primarily used for linear video, high-density interactive set-top and desktop CD-ROMs drives will eventually replace the ubiquitous double and quad-speed drives as DVD or HDCD drives become viable commercial products.[39] The technology to support the demands of full motion featured films on CD will improve the synchronization of concurrent heterogeneous media components (i.e., QuickTime movies with several media tracks). This is a common issue associated with multimedia development and better interactive environments will not be available for the desktop and set-top CD-ROM systems until it is resolved.

5.4 Discussion

A presentation scheme or format is only successful if it is useful or appropriate for mass consumption. Most interleaved formats have long disappeared due to compatibility problems and the requirements for peripheral devices to view and hear the accompanied data. The Video CD format is unsupported by many desktop and set-top platforms because it does not meet the requirements (i.e. the quality of MPEG-1 video, and limited playback time) set out by the Hollywood Digital Video Advisory Group (HDVAG) as a standard video format for motion pictures.[40] High Density CDs are still in development and until they become standard formats, their use for interactive music CDs is still unknown.

The Enhanced audio CD format is the most impressive and commercially viable presentation scheme for the multimedia and audio CD market. It is the only format that allows end-users to play linear audio and multimedia content on several platforms. Its commercial success is evident in the growing number of Enhanced audio CDs that are presently available.

Chapter Six
Desktop Music CD-ROMs

6.1 Introduction

Desktop music CD-ROMs evolved from Warner Bros. Records' interest in providing extended media (i.e., video, graphics, and text) for an existing audio CD. This may be attributed to the fact that virtually all computers now come equipped with built-in CD-ROM drives, high resolution monitors, sound cards with built-in General MIDI sound samples, and sophisticated authoring and production tools for the integration of digital media components for a CD-ROM music presentation.

The introduction of interleaved and mixed mode formats in the late 1980s sparked a growing interest among members of the recording industry, recording artists, and musicologists, in developing multimedia presentations with an emphasis on music for CD-ROM. Just as the introduction of music videos in the early 1980s produced a new commercial genre, music CD-ROMs are becoming their own legitimate form of expression. Also, due to the fact that there is a clear demarcation of musical styles and distinct interpretations of how media is to be presented, a wide variety of music CD-ROMs are presently available. Categories include music appreciation or reference works, retrospectives, interactive music presentations, and multimedia content as an accompaniment to linear audio based on the Enhanced CD format.

6.2 Music Appreciation: The CD Companion

Music appreciation CD-ROMs serve to provide general information about a specific composer, music history, and rudimentary music to a wide audience of both musicians and non-musicians. The *Ludwig Van Beethoven: Symphony No.9 CD Companion* was the first consumer music product that integrated linear-audio with multimedia data based on the mixed-mode format. The CD presented the full recording of *Beethoven's 9th Symphony*, and allowed the participant to navigate through the presentation to read biographical information, view photographs of the composer and sketches of the original scores, listen to musical excerpts (often accompanied with music notation), read about significant events within the composition, and to view other musically relevant material the click of a mouse.

The two primary publishers of music appreciation CD-ROMs include Time Warner Interactive and Voyageur. Time Warner Interactive's *Audio Note Series* includes several music appreciation CDs : *Beethoven String Quartet No. 14, Johannes Brahms: A German Requiem, B. Britten's The Young Person's Guide to the Orchestra,* and Mozart's Opera*The Magic Flute.* The Voyageur *CD Companion Series* includes: *Igor Stravinsky: The Rite of Spring, Wolfgang Amadeus Mozart: The "Dissonant" Quartet, Antonin Dvorak: Symphony No.9: From the New World, Franz Schubert: The Trout Quintet,* and *Richard Strauss: Three Tone Poems.* The fact that these presentations were developed and marketed for a wide audience of musicians and non-musicians is the reason that they have become some of the most popular CD-ROMs presently available — in *any* category.

6.3 Retrospectives

Another significant music CD-ROM type is the retrospective, often called a compendium. Retrospective music CD-ROMs place emphasis on a particular artist (*Xplora1: Peter Gabriel's Secret World, John Lennon: Imagine, Bob Dylan CD-ROM, Virtual Graceland, [Prince]: Interactive*), group (*Heart: 20 Years of Rock & Roll, Yes: Active*), work (*The Who's Tommy: An Interactive Journey*), or event (*Woodstock: 25th Anniversary*) and function as a multimedia database of music, video, graphics, text, biographies, backstage scenes, personal information, interviews, live performances, games, and basic music theory.

Xplora1 was released in 1993, and marked the first retrospective music CD-ROM by a recording artist. This CD-ROM contains information about Peter Gabriel, his new album *US* with video clips and paintings associated with each song, a complete discography with sample excerpts and lyrics, information about his involvement with the various political and arts organizations, a behind the scenes look at Real World Studio, and a vast array of 'world' instruments playable by the interactive control of a mouse click. *Xplora1*, like many other music compendiums, is based on the notion of exploration of different environments. These titles have been commonly referred to as "exploratoriums." [41] In comparison with later music retrospectives, *Xplora1* still remains the most impressive with respect to content, design, and technology.

Most of the information and media on a CD compendium is already accessible through record stores, MTV, Much Music, and the World Wide Web. The distinction, however, lies in the distribution medium (i.e. one CD), but whether the public is willing to pay $49.00 per CD remains an open question.

6.4 Personal Music Theatre

The term "Personal Music Theatre" is used to identify an environment in which an author (composer) provides the "Media Lego Blocks" (i.e. audio, motion video, graphics, media processes, and descriptors) to allow end-users to customize personal listening and viewing experiences, and to take on role-playing situations including conductor, performer, composer, sound engineer, music-video producer, and audience/listener.[1] The aim of such an environment is to allow the end-user to make choices in the creative aspect of a presentation on CD-ROM.

Interactive media technology has allowed many prominent recording artists, "new media" artists, interactive divisions of media industries, and contemporary music composers to create interactive presentations based on the personal music theatre theme. Peter Gabriel is one artist who has attempted to blur the distinction between artist and non-artist through interactive media technology, by allowing end-users to express themselves as "collaborators" in the presentation. [42]

The music presentations to be discussed in the following section allows for creative input from the end-user.

6.4.1 End-User as Artist

Musical systems are being developed to allow one to explore a set of musical parameters, assemble musical segments, customized personal listening and viewing experiences, perform on virtual instruments, and interact with digital audio and MIDI tracks.

For example, Paramount Interactive's *Rock, Rap and Roll* was designed to introduce an audience to different styles of popular music. It is an interactive virtual 'jam session' in which the participant takes an active role in adding their own parts to a series of pre-composed audio segments (generally consisting of four-bar repeating rhythms with accompanied chord progressions). This highly interactive presentation allows one to perform with and listen to a diverse selection of musical styles including: Reggae, Jazz, Rap, Rock, African, Big Band, Soul, Latin and others.

In the role of the performer, the participant chooses a musical style to interact with by arranging twelve unique four-bar music segments (represented as graphic "music coins") in a "virtual jukebox". To interact with the concurrent audio tracks, the participant uses the mouse (used to trigger samples), and the computer keyboard —where each key is "mapped" onto a particular sampled sound (built-in and user

[1] A term coined by Prof. Bruce Pennycook. "Interactive CD Proposal." Manuscript. 1994.

defined) stored on the CD-ROM or local hard disc. In essence, the computer keyboard becomes a new interactive musical instrument for non-musicians to compose.

6.4.2 Customized Music and Performer's Choice

Todd Rundgren has explored the possibilities of interactive media through his re-programmable audio CD concept. Rundgren supplies the pre-defined musical tracks and the tools that allow the end-user to customize or reprogram the content.

In 1993, Todd Rundgren introduced *TR-I: No World Order* as an alternative/enhancement to his linear audio-CD version of the same title. *No World Order* is loosely based on the same concepts employed in the compositions of Boulez, Stockhausen, Cowell, and Cage, yet it is performed on a CD-based desktop computer, rather than in a concert hall setting.[2]

According to Rundgren, *No World Order* consists of "musical agents" that have an intimate knowledge of all the musical data stored on the CD-ROM. [43] The agent attempts to make intelligent choices based on the parameters that have been specified by the end-user. The CD-ROM is essentially a database of approximately 933 four-and-eight second musical segments, scripts, and buttons that allow one to modify and customize the pre-existing stored material. There are seven "flavor" settings—program, direction, form, tempo, mood, mix, and video, which are then sub-

---

[2] Customized music experiences has existed since the 18th century with the inception of Mozart's "dice music". Dice music was based on a system that determined the arrangement of musical phrases and form by the role of a dice. This effective tool for non-composers and artist to create personalized music based on pre-existing musical phrases, was considered to be a simple diversion and compositions created with this system were seldom performed.

Twentieth composers such as Boulez, Stockhausen, Earle Browne, and Cage introduced components of indeterminacy with regard to composition and performance. The terms aleatoric, chance music, and music of indeterminacy have been associated to compositions based on the notion of chance. In performance, chance is allowed to operate by leaving some elements and/or their order of appearance to the performer's discretion. Karlheinz Stockhausen's *Klavierstuck XI* (1956) is based on the notion of performer's choice. The work permits the pianist to choose: 1) the order of which the nineteen sections are to be performed; and 2) from six distinct tempos and dynamics. Another work which includes the concept of performer's choice is *Available Forms for Orchestra*, by American composer Earle Browne. The form of the composition is determined by the conductor and performer based on a "map" indicated by the composer that is used to determine the form of the piece. In *String Quartet No. 3* (1935) by Henry Cowell, require that the performer the re-assemble various musical segments at their discretion. Another form of chance can be found John Cage's *Music of Changes* for piano (1951). In this piece the duration and frequency of occurrence of pre-determined pitch aggregates were dictated by the *I-Ching* (Book of Changes). These concepts and processes subject a musical work to partial or total transformation from one performance and performer to another.

categorized even further that are used to control how the segments can be played, controlled, and ordered.

No World Order is a very interesting concept and is the first re-programmable CD. It is unclear whether other prominent recording artists will also explore the possibilities it raises; the idea of re-assembling music segments to personalize a musical experience may be too complex and esoteric for non-musicians. Thus, recording artists and serious composers may be reluctant to implement an unfinished work, song, or album.

6.4.3 Audio Engineer

Xplora1 features a virtual mixing console that is used to re-mix "Digging in the Dirt." This particular song contains four tracks (guitar, bass, drums and vocals) in which the end-user can control the volume and save the re-mixed version as an external audio file. The volume of individual QuickTime music tracks is controlled by manipulating "virtual sliders" displayed on screen using the mouse.

This is the first and only interactive CD-ROM that allows end-users to re-mix digital audio in real-time but unfortunately, the quality of digital audio and the response time per track is quite poor. This is due to the limited transfer rates of a double-speed CD-ROM drive. An alternative to using digital audio for real-time mixing from CD-ROM would have been to use a multitrack Standard MIDI File (SMF). This would provide immediate response times and potentially better audio quality than an audio file encoded at 8-bit, 11kHz mono. Appendix 6.2 illustrates several HyperTalk scripts that use HyperMIDI external commands and functions (XCMDs and XFNCs) to play and control the volume of individual tracks of a multitrack Standard MIDI File (SMF) from a "virtual mixing console."

6.4.4 Music Video Producer

Since the early 1980s, music videos have become as important to the recording artist as the audio CD. With the advent of music video stations including MTV and Much Music, artists have realized the importance of making music videos as a new distribution medium. Recent interactive music CD-ROMs including David Bowie's *Jump: The David Bowie Interactive CD-ROM* and Pop Rocket's *Total Distortion* present the participant with the illusion of being immersed in an interactive MTV by allowing them to take on the role of the music video producer.

Bowie's *Jump: The David Bowie Interactive CD-ROM* presentation is essentially an audio and video editing product containing interactive events triggered by various graphic objects. Based on David Bowie's *Jump They Say* music video, end-users are provided with five different reels of video footage, a five-channel video mixer, and two unique audio recordings of the same song. The material can be re-assembled, edited, and finally produced according to the participant's desires.

Total Distortion is as an "interactive music video adventure." [44] It blends elements of interactive music, video mixing, adventure games, and virtual exploration. The end-user takes on the active role of a music video producer in search of new material to create original clips. In the "Personal Media Tower," one selects from various musical excerpts and images to create a personalized music video. The music video producer room is the main feature of this title, however, there are other environments with which one can interact.

6.4.5 Virtual Instruments

Virtual instruments in the context of music CD-ROMs are essentially a graphic representation of an instrument plus a set of sampled instrumental sounds. Virtual instruments can be found in music CD-ROMs including the world music section of *Xplora1* , *Musical Instruments*, and *B. Britten's The Young Person's Guide to the Orchestra*. These instruments are nearly impossible to perform. This is mostly due to slow response and access time from CD-ROM, and the akwardness of a mouse or keyboard as a controller.

Virtual instruments based on the QuickTime General MIDI sound samples have not been fully explored in interactive CD-ROM music presentations. This may be attributed to the fact that it is still not possible to add user-defined sound samples to the QuickTime Musical Instruments Extension. Once this problem is resolved and the sound quality has improved from 8-bit to 16-bit audio, QuickTime General MIDI virtual instruments may offer intruiging possibilities for future interactive music environments.

6.5 Enhanced Audio CD Presentations

As discussed previously, Enhanced CDs are based on the multi-session presentation scheme for adding multimedia content to linear audio CDs. The audio is the primary reason for the CD to exist, and for some artists, the added material serves to complement the audio. For others, it serves as an interactive audio and video experience.

Artists such as Sarah McLachlan and Mike Oldfield have created CDs featuring this new technology. [45] The most impressive use of this technology is found in The Resident's *Gingerbread Man* Enhanced CD presentation. It is based on the inner thoughts of nine 3D photo-realistic characters that are "linked" or associated with nine musically related songs. The audio is linear and fixed: the interactivity surfaces when the end-user "plays" the characters. This is achieved using the computer keyboard and mouse to trigger effects that change facial expressions, audio samples, and images. Unlike other enhanced albums that offer biographies, liner notes, and music videos, this is the first expanded album that offers a visual interactive experience tied directly to a full-length audio CD. [46]

6.6 Discussion

The level of user interaction and the control over media components in most interactive music CD-ROMs is quite primitive compared to what can be created with programming environments such as Max. This is may be attributed to the delivery platform (CD), the lack of sophisticated authoring tools for innovative music presentations, and the media used.

Also, there are very few interactive music CD-ROM presentations that use MIDI data as the source of audio. This type of data can be easily transformed or re-structured and is not as data intensive as digital audio. Its use would increase the amount of audio tracks and improve the response time for interactive events.

As a majority of music CD-ROMs are developed using authoring tools such as HyperCard and Director (fast development time), the only means of programming interactive "music engines" is the HyperMIDI external functions. This library of MIDI commands pales in comparison to to these offered by Max, and until compiled Max patches can be integrated with sophisticated authoring tools, most music CD-ROMs will be limited in their interactivity.

Chapter Seven
Gaming Environments

7.1 Introduction

Game developers are primarily concerned with visual presentation, interaction between computer and end-user, and fast response times. Since these are also essential components for networked or CD-based interactive music systems, it is imperative that the computer music research community learn from gaming environments.

7.2 Computer Networked Games

The communication protocols of networked games could be important models for the specifications and ultimate design of networked music systems. These games can be divided into two categories: non real-time and real-time multiplayer environments.

7.2.1 Non Real-Time Games

Computer network games have existed since the development of electronic mail. The first games were not in real-time and consisted of "play by mail interactive fictions" over modems. With the World Wide Web's support for hypermedia, there has been a strong growth of interactive games (Web-Chess, The Virtual Slot Machine, Web-Tris, Web-Yahtzee) and interactive fictions (WAXweb and MUDs) available over the Web.[47][48][49] There are also an increasing number of interactive games being written in the Java programming language as executable applets.[50]

Although these diversions do not offer real-time interaction between connected end-users over a network structure and were not designed to support multiplayer interaction, they are nonetheless significant in that they provide intuitive interfaces, and multi-user interaction. These are important features for music systems.

7.2.2 Real-Time Games

MUDs (Multi-User Dungeon) were the earliest real-time text-based virtual reality environments over computer networks. Since then, computer networked games have evolved into truly interactive multiplayer environments that can support 3D rendered graphics and digital audio.

Doom, *Marathon*, and *Bolo* are network strategy war games that are similar to typical arcade-style "shoot-em-up" action environments. Unlike Web games which rely on HTML protocols, all three were written in the C programming language and use real-time communication protocols (UDP/IP multicast). These two factors allow these games to function in real-time. In Bolo, players can participate as a single IP player or as a member of a team over the Internet. As shown in Figure 7.1, network communication methods used include 1) single player (against the computer); 2) serial port communication (computers are connected via the modem or printer ports); 3) AppleTalk (over LocalTalk or as EtherTalk over an Ethernet system); 4) UDP/IP (connectionless Transport Layer protocol that offers un-reliable fast data transmission to an IP address); and 5) UDP/IP multicast (similar to the above protocol, however, one packet is distributed to participated players enrolled in a *Bolo* game). (See Figure 7.1) one packet is distributed to participated players enrolled in a *Bolo* game). (See Figure 7.1)

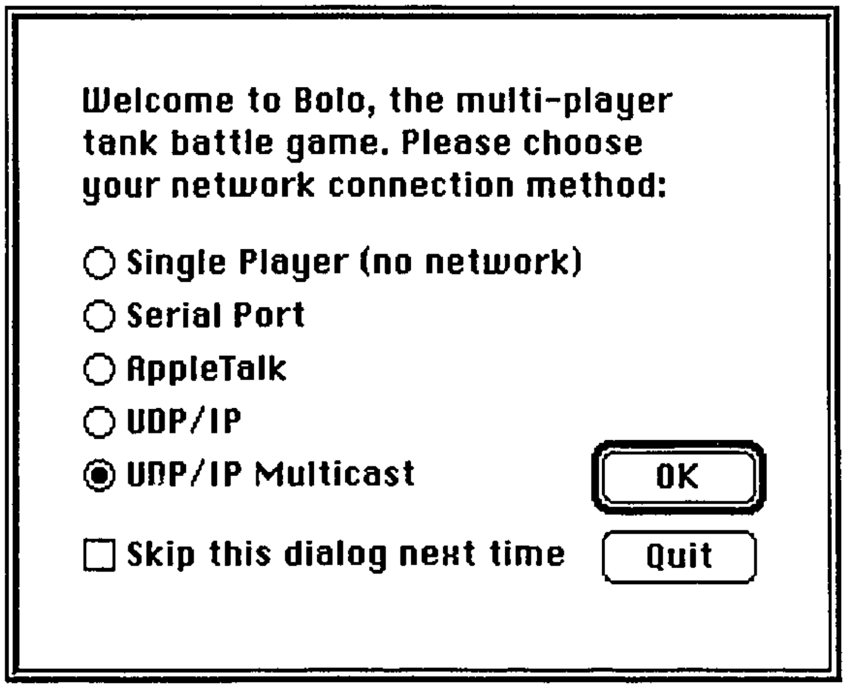

Figure 7.1: Bolo offers five network connection possibilities.

The significance of real-time networked games is that their communication protocols allow real-time interplay between multiple players. Control information rather than digital media is transmitted over a networked structure allowing these games to function in real-time.

7.3 CD-Based Gaming Environments

CD-based computer games provide the end-user with fast response times, interactive branching, stunning 3D rendered images, complex strategies, various levels of difficulty, gaming intelligence, and CD quality sound effects. On the other hand, game developers are not concerned with creating a dynamically changing musical environment but rather they use music merely as a static accompaniment or backdrop to the action scenes.

In games music serves several purposes such as sound effects, ambiant music, and transitions between settings. This is the case in interactive fiction and role-playing environments developed for CD-ROM (*Myst*, *7th Guest*, *The 11th Hour*, and *The Journeymen Project*) and the majority of set-top games.

The implementation of processes that modify musical parameters (tempo, dynamics, key, pitch, rhythm, timbre, articulation etc.) and audio parameters (e.g. acoustic space) in order to create auditory experiences that are tied to the action scenes, could have an impact on the end-users level of interest during the presentation or game. These processes are extremely important in Virtual Reality (VR) systems in which body movements are often used to modify digital media parameters during a presentation.

7.3.1 Set-top Gaming Environments

Games are the most prominent of all CD consumer products developed for set-top platforms. As these platforms have improved to support CD-audio, there have been virtually no attempts to enhance a game through music. In fact, music has remained a nearly insignificant component of the overall presentation. This may be attributed to the types of applications created, which do not require sophisticated audio processing, intriguing music, or the transformation of musical parameters. Also, set-top systems do not support MIDI capabilities. As a result , the implementation of interactive "music engines" based on MIDI to modify musical parameters is impossible at the moment.

If these processes were to be implemented in computer games, they would have to be transparent to the user and not interfere with interaction with the game.

7.4.2 Desktop Gaming Environments

Music in desktop gaming systems functions in a manner similar to that of set-top games. However, since desktop games are developed to be played on personal computers which support MIDI capabilities, interactive "music engines" based on MIDI can be implemented easily. In fact, algorithmic music patches created in Max could be imported into the gaming environment as compiled applications. Furthermore, QuickTime VR technology is featured in the *Star Trek: The Next Generation Interactive Technical Manual*, desktop CD-ROM game. The game was created entirely from 35-mm photographs, and permits end-users to navigate around the "Virtual Enterprise" in a 360-degree perspective. The CD-ROM also allows the participants to "pick up" objects and trigger events by clicking the mouse over a "hot-spot" region of the QuickTime VR interface.

Using QuickTime VR and Max together has not been explored fully in a commercial CD-ROM presentation and is therefore difficult to predict the combined effect on interactive music for desktop gaming environments.

7.4 Discussion

The impact of enhancing an overall gaming presentation by the manipulation of auditory and musical parameters is still unknown. A cross-disciplinary collaboration between CD-based gaming developers and the computer music research community could in fact have a major impact on how music is applied to CD-based applications for set-top and desktop systems.

The computer music community has been involved in advanced research areas such as music perception, music cognition, artificial intelligence (AI), neural networks, genetic algorithms, interactive performance and composition music systems, expert systems, pattern recognition, virtual instruments and acoustics, audio signal processing, real-time sound synthesis techniques, 3D audio, networked music systems, and collaborative music environments through Web-MOO's. Yet very little of this research has surfaced in mass market applications. In fact, the computer music community in general seems relatively isolated from the computer gaming and interactive music CD-ROM industry. It seems that most of the work is focused on contemporary and experimental music compositions and interactive performance pieces whereas the primary focus of CD-ROM applications is to produce immediately accessible media for a mass market.

Only when these two disciplines collaborate will better music be provided in gaming environments.

Only when these two disciplines collaborate will better music be provided in gaming environments.

Chapter Eight
Hybrid CD-ROM/ Networked Systems

8.1 Introduction

Web browsers such as Netscape 2.0 and HotJava are transforming Web pages into interactive media environments. Both browsers require that the media be downloaded before an end-user can interact with the presentation. In most cases the medium over which these applications propagate is a 14.4K or 28.8K modem. As a result, applications that are data intensive may require excessive download delay times. This will cease to be an issue, however, as transmission mediums improve to support real-time interactive presentations. At the moment a temporary solution is a hybrid system of CD-ROM and networking technologies. A hybrid system offers fast data access and response times, a large storage medium for applications and media components, and dynamic multimedia content.

8.2 Hybrid Applications

This section will outline four possible hybrid CD-ROM/networked systems for music: 1) networked multimedia extensions to a linear audio CDs; 2) Web documents as interfaces to local media stored on a CD-ROM; 3) networked agent-based personal music recommendation systems; and 4) multi-user networked systems.

8.2.1 Networked Multimedia and Audio CDs

Recording companies including Sony, MCA, Geffen, and Atlantic records have been creating Web documents for recording artists as a complement to newly released linear audio CDs [51][52][53][54]. These documents include biographies, photographs, music videos, musical excerpts and other related material. At present, there is no direct connection between the audio CD and the multimedia content found on the Web. There is also no means of accessing both content from one medium. The ability to access multimedia content over a network structure from a CD-ROM or an audio CD would provide the end-user with updated digital media: a networked version of the Enhanced CD format.

Other possible uses of this type of hybrid system could include: 1) a connection to a *newsgroup* , *mailing list* , or a Web page devoted to a particular artist; 2) a *telnet* session to an on-line music library; 3) on-line ordering service for new CDs and other related materials; and 4) a connection to a Web-MOO database whereby consumers could connect to a "virtual chat room" to discuss issues regarding the artist and the content with other on-line consumers. (See Figure 8.1)

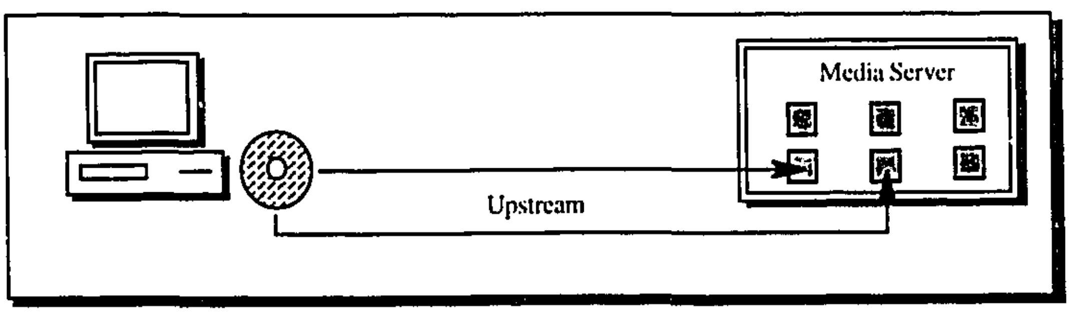

Figure 8.1: CD-ROM functions as launch utility for on-line forums and extended multimedia content.

Establishing a connection between a Web browser such as Netscape with an external application residing on a local CD-ROM may be implemented on Macintosh hardware through AppleEvents. Netscape uses AppleEvents to interact with other Macintosh applications through scripting languages such as AppleScript or any programming language. [55] Appendix 8.1 shows an example HyperTalk script used to launch the Netscape Web browser with a specified URL.

8.2.2 Web and CD-ROM

Numerous hypermedia documents on the Web contain data intensive media components. At times these are important elements in the semantics of the presentation, yet due to excess download delay times, they are disregarded. A Web document as an interface to the local media on a CD-ROM provides better response times and presentation flow. (See Figure 8.2)

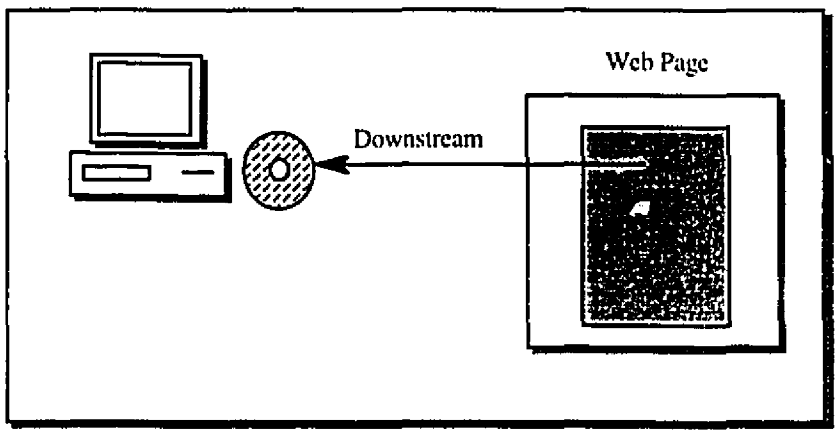

Figure 8.2: Web as the interface to local media on CD-ROM

This system could be implemented using AppleScript scripts and the *Flypaper* communication utility [56]. Flypaper provides a link between anchors on a Web page and AppleScript scripts residing on a Macintosh or CD-ROM. Flypaper executes the AppleScript script that is associated with the "href" link. Since the scripts are pre-compiled and residing on the local medium, the execution of the digital media is immediate. (See Figure 8.3)

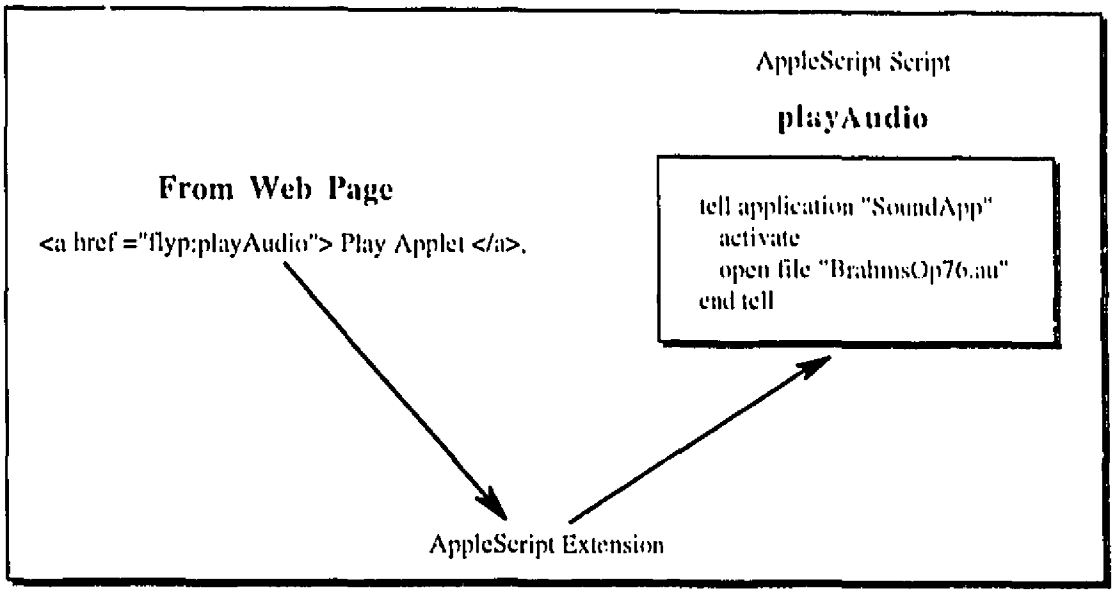

Figure 8.3: Illustrates a hybrid system where the Web page functions as an interface to a compiled AppleScript application. Clicking on the "Play Applet" anchor executes the "playAudio" script. The playAudio script must reside in the same folder as the Flypaper application. The "flyp" is the flypaper protocol that is used to execute the appropriate script on a local medium.

This hybrid system does not require the end-user to download the digital media components that are referenced by the application.

8.2.3 Navigating with Intelligent Agents

The field of intelligent agents has become a "hot" research topic for many computer science cognitive science, and computer music researchers.[57] Intelligent agents have been applied to the Internet to provide assistance to an end-user who requests specific items or media according to a pre-defined set of preferences. Figure 8.4 illustrates a possible hybrid model where scripts on the CD-ROM could dispatch agents to search various media servers for updated digital content. This concept is similar to the ideas implemented in *Firefly*. [58]

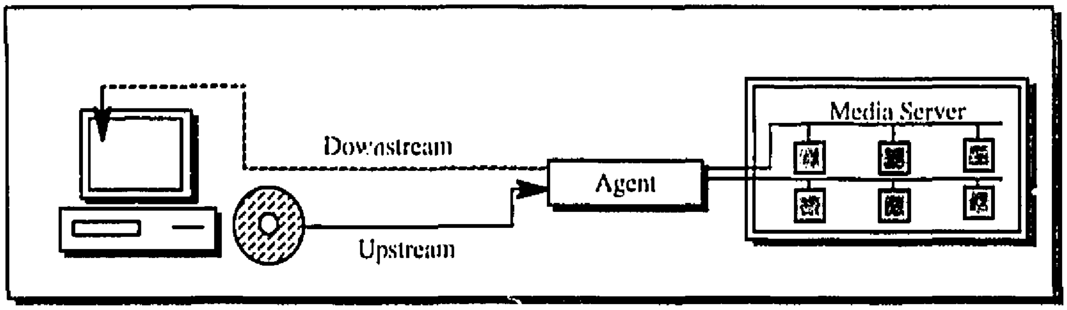

Figure 8.4: Hybrid system using intelligent Agents.

Firefly is a personal music recommendation agent that searches the Web for hypermedia documents of a particular artist and musical styles according the end-users musical preferences. The interface presents various musical styles and artists, and the end-user rates the presented styles according to a rating system (i.e., 1=bad and 10=good). The music agent connects the user to a URL that contains pointers to all of the hypermedia documents related to a particular artist or musical style that the user rated the highest.

8.2.4 Hybrid Multi-User System

The interaction within the above hybrid systems is essentially point-to-point. Figure 8.5 illustrates a multi-user system of connected computer systems to a remote media server. The local media on the CD-ROM is controlled by other participants connected to the system.

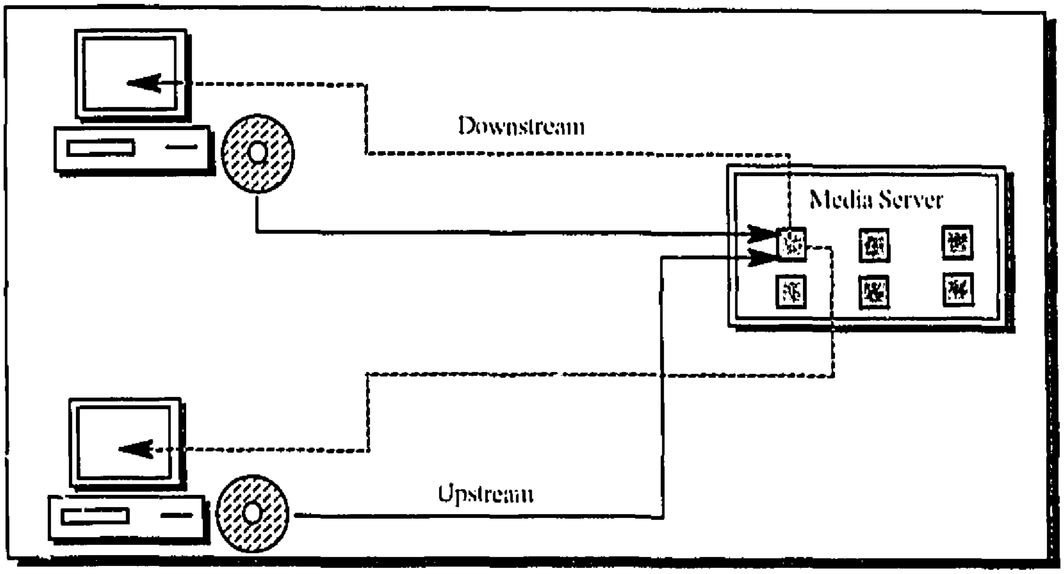

Figure 8.5: Illustrates a hybrid multi-user interactive environment

The server functions as a "media manager" and routes control commands to the appropriate client CD-ROM. This hybrid system could be used in multiplayer game environments and distributed multimedia presentations controlled by a remote server.

8.3 Discussion

Creating a server for a music environment based on a hybrid system is extremely complex. The server needs to keep track of the media and to ensure that events are synchronized and coherent for all users. Furthermore, in a role-playing musical environment where a client may take on the active role of performer, conductor, composer, or audio engineer, a server would not only need to interpret the meaning of these roles, but to structure the digital media accordingly. Such a server could be modeled after the *Habitat* project from Lucas Films [59]. Habitat is an on-line environment in which players adopt an animated persona, or "avatar," that communicates and interacts with other avatars. The server monitors and executes object positions (transmitted over a 28.8 Kbps modem), and controls graphics that are located on a CD-ROM.

A hybrid CD-ROM and on-line connection may be a short term technology. But until interactive broadband networks arrive, there will likely be plenty of intriguing hybrid experiments.

Chapter Nine
Conclusion

This thesis has described the elements for the development of interactive music systems for CD-ROM and networked systems. It has further demonstrated that an improvement in communication protocols, transmission mediums, CD-ROM technology, and digital media authoring tools is necessary before seemless, full-bandwidth interactive music systems can be fully implemented.

The growing number of mulitmedia applications (e.g. RealAudio, CU-SeeMe, MBone, HotJava, Netscape and *Xplora1*) that have surfaced in recent years provide exciting glimpses of what is to come for interactive media over networked and CD-ROM systems. It is certainly clear that many artists and software developers have become interested in creating systems that allow end-users to take part in the creation of a work. This concept has inspired a new art form that allows the end-user to become the artist. New systems are providing the end-user with primitive tools (i.e. the mouse and computer keyboard) to compose, conduct, perform, re-arrange, and customize music according personal preferences.

Just as computing, television, and radio have evolved over the years to meet the demands of new applications and concepts, CD-ROM and networked interactive multimedia applications will also continue to evolve as technological innovations provide better control over media, real-time data transmission, and reliable communication protocols.

Appendix 2.1
MIDI

MIDI overview and specification

MIDI Messages

Channel Messages

Channel Messages use the lower 4 bits of the status byte to indicate the MIDI channel of the message. 0 is MIDI channel 1, and 15 is MIDI channel 16.

Function	*Objects*	*Status Byte* Decimal	Hex	Binary	*2nd Byte* (0-127)	*3rd Byte* (0-127)
Note Off	xnotein, xnoteout	128-143	80-8F	1000xxxx	Key Number	Release Velocity
Note On	notein, noteout	144-159	90-9F	1001xxxx	Key Number	Velocity
Poly Pressure	polyin, polyout	160-175	A0-AF	1010xxxx	Key Number	Aftertouch
Control Change	ctlin, ctlout	176-191	B0-BF	1011xxxx	Controller Number	Controller Data
Program Change	pgmin, pgmout	192-207	C0-CF	1100xxxx	Program Number	
Aftertouch	touchin, touchout	208-223	D0-DF	1101xxxx	Aftertouch Value	
Pitch Bend	bendin, bendout	224-239	E0-EF	1110xxxx	Bend (LSB)	Bend (MSB)
System Messages						
System Exclusive	sysexin, midiout	240	F0	11110000	Mfgr ID Number	Arbitrary
Song Pos Ptr	midiin, midiout	242	F2	11110010	Position (LSB)	Position (MSB)
Song Select	midiin, midiout	243	F3	11110011	Song Number	
Tune Request	midiin, midiout	246	F6	11110110		
End of Sys Ex	sysexin, midiout	247	F7	11110111		
Clock	rtin, midiout	248	F8	11111000		
Start	rtin, midiout	250	FA	11111010		
Continue	rtin, midiout	251	FB	11111011		
Stop	rtin, midiout	252	FC	11111100		
Active Sensing	midiin, midiout	254	FE	11111110		
System Reset	midiin, midiout	255	FF	11111111		

Taken from p.238 of the Max manual. Opcode Inc.,

Appendix 4.1

Lingo and HyperMIDI XCMDs used to control graphic objects.

This handler initializes 9 MIDI controller streams and makes them into internal objects with buffers, etc. Further below is the Factory for MidiNote, one of those internal controller objects. Each controller requires its own factory.

```
on initialiseMidi
  global noteStream,controller1Stream,controller2Stream,aftertouchStream
  global pitchBendStream,footPedalStream,midiInUse,midiThru
  global lowestpitch,pitchInterval,volumeStream,progChangeStream
  if midiInUse then
    hmOpenMIDI "8000 8000 8000 8000 8000 8000 8000 8000 8000",8000
    hmPatcher "connect","HMid,1in1,input,amdr,Ain ,"
    hmPatcher "connect","HMid,1ou1,output,amdr,Aout,"
    hmWriteMIDI 1, "144,36,100, 200:144,53,100, 400:144,60,100,
600:144,36,0,800:144,53,0, 1000:144,60,0"
    set noteStream=MidiNote(mNew,list(1,2,3,4))
    set controller1Stream=MidiController1(mNew)
    set controller2Stream=MidiController2(mNew)
    set aftertouchStream=MidiAftertouch(mNew)
    set volumeStream=MidiControllerVolumeStream(mNew)
    set pitchBendStream=MidiPitchBendStream(mNew)
    set footPedalStream=MidiController4(mNew)
    set progChangeStream=MidiProgramChange(mNew)
  end if
end initialiseMidi
```

---

this method is part of a factory that controls various graphic objects with incoming MIDI data. This one in particular assigns incoming note and velocity MIDI data to the instance variables noteValue and velocityValue. These are used in other methods to control objects (their position, speed, colour, castmember, etc - in fact anything that can be controlled with Lingo). Managing the MIDI buffers so that any mis-timings between the Mac and the source are accounted for, thus we never lose data.

```
method mNoteVelocity
  global noteStream
  if noteStream(mNewValueReceived) or not noteStream(mLatestEventHasBeenUsed)
then
    if noteStream(mLatestEventHasBeenUsed) then
      set noteValue=noteStream(mReadLatestPitch)
      noteStream(mSetLatestEventAsNotUsed)
    else
      set noteValue=noteStream(mPitch)
    end if
    set chanReceived=noteStream(mChannel)
```

```
    if chanReceived=channel then
      noteStream(mSetLatestEventAsUsed)
    end if
    set velocityValue=noteStream(mVelocity)
  end if
  if not noteStream(mIsChanInUse,chanReceived) and not
noteStream(mLatestEventHasBeenUsed) then
    noteStream(mSetLatestEventAsUsed)
  end if
```

---

this Factory is where all data to do with the controller handling note and velocity (they are sent together in MIDI) information is managed. The first few methods are all concerned with buffer management and the later methods are concerned with hexadecimal conversion of MIDI type data to decimal values in the range 0-127. These decimal values are then assigned, in the previous method, to variables which are used to control Director events.

```
factory MidiNote

method mNew chansInUseList
  instance note,latestEventHasBeenUsed,chansInUse
  set chansInUse=chansInUseList
  hmSetFilter 1,"block all, pass noteOn"
  set latestEventHasBeenUsed=true

method mLatestEventHasBeenUsed
  return latestEventHasBeenUsed

method mSetLatestEventAsUsed
  set latestEventHasBeenUsed=true

method mSetLatestEventAsNotUsed
  set latestEventHasBeenUsed=false

method mIsChanInUse chan
  return getPos(chansInUse,chan)<>0

method mNewValueReceived
  return (hmUtility("getInputCount",1)>0)

method mReadNextPitch
  me(mReadNextNoteOn)
  return me(mPitch)

method mReadLatestPitch
  me(mReadLatestNoteOn)
  return me(mPitch)

method mReadNextNoteOn
  set note=hmReadMidi(1,1,"nostamp")

method mReadLatestNoteOn
```

```
 repeat with i=1 to ((hmUtility("getInputCount",1))-1)
  hmReadMidi(!,1)
 end repeat
 me(mReadNextNoteOn)

method mPitch
 return hmConvert("extract",note,3,1,1)

method mVelocity
 return hmConvert("extract",note,3,2,0)

method mReadChannel
 me(mReadLatestNoteOn)
 return me(mChannel)

method mChannel
 return hmConvert("extract",note,3,0,2)-143
```

Appendix 6.1

Read SMF

```
-- reads a MIDI file into the global variable theSeq
on mouseUp
 global theSeq
 get filename("Midi")
 if it is not empty then
   go to this card
   set cursor to watch
   put empty into theSeq
   put hmMIDIfile("read",it,"msec") into theSeq
   -- check for errors
   if char 1 to 5 of theSeq is "Error" then
     answer theSeq with "OK"
     put empty into theSeq
   else
     checkBufSize
   end if
 end if
end mouseUp

-- this routine sees if theSeq will fit in the output buffer,
-- and, if not, create a larger output buffer by reopening HyperMIDI
on checkBufSize
 global theSeq
 -- see if sequence will fit in buffer
 hmOpenMIDI "!"
 put line 3 of the result into bufSize
 hmWriteMIDI 1,theSeq,"count"
 put the result into requiredSize
 if bufSize < requiredSize then
   -- kill any playing sequence
   if hmUtility("getOutputCount",1) > 0
   then get hmUtility("killOutput",1)
   -- reopen will larger buffer
   hmOpenMIDI 400,requiredSize
   if the result is not empty then answer the result with "OK"
   hmPatcher "connect",cd fld "connections" of cd 1
 end if

end checkBufSize
```

PlaySMF

```
-- does nothing if we're already playing
on mouseDown
  -- only play if we're not already playing something
  if hmUtility("getOutputCount") is 0 then
    global theSeq
    set cursor to watch
    set hilite of me to true
    hmWriteMIDI theSeq  -- start playing
  end if
end mouseDown
```

Pause SMF

```
on mouseUp
  if hilite of me is false then
    set hilite of me to true
    get hmClock("stop")               -- stop the clock
    get hmUtility("muteOutput",1)    -- turn off any sustaining notes
  else
    set hilite of me to false
    get hmClock("start")             -- restart the clock
  end if

end mouseUp
```

Mute SMF Track

```
on mouseDown
  if hilite of me is false then
    set hilite of me to true
    -- shut the bass,drums and solo off
    get hmUtility ("setMutes",1,"1000000000000000")
  else
    set hilite of me to false
    get hmUtility ("setMutes",1,"0000000000000000")
  end if

end mouseDown
```

Solo SMF Track

```
on mouseDown
  if hilite of me is false then
    set hilite of me to true
    get hmUtility ("setMutes",1,"0111111100000000")
    repeat with i = 1 to 8
      disable bkgnd button ("mute" &&i)
    end repeat

  else
```

```
    set hilite of me to false
    get hmUtility ("setMutes",1,"0000000000000000")
    repeat with i = 1 to 8
      enable bkgnd button ("mute" &&i)
    end repeat
  end if
end mouseDown
```

SetLevels (User Defined)

```
-- set all volume faders:
-- if option is held, set to full
-- if option-shift, set to level of fader 1
-- else query
on mouseUp
  if the optionkey is down then
    put 127 into level
    if the shiftKey is down then
      put line 4 of cd fld "Vol 1" into level
    end if
  else
    ask "Set all faders to:" with 127
    if it is empty then exit mouseUp
    if it > 127 then get 127
    else if it < 0 then get 0
    put it into level
  end if
  set lockscreen to true
  put cd fld "curVols" into temp
  repeat with i = 1 to 8
    set cursor to busy
    -- change the slider position
    -- put new values in each fader and send the message
    put level into line 4 of cd fld ("Vol"&&i)
    hmSlider id of cd fld ("Vol"&&i),0,"sendSet"
    -- if channel not muted, update list of volumes
    if hilite of bkgnd btn ("mute"&&i) is false
    then put level into word (i*3) of temp
  end repeat
  hmWriteMIDI temp
  put temp into cd fld "curVols"
end mouseUp
```

APPENDIX 6.1- (INITIALIZE VOLUME CONTROLS) pg 89

Appendix 8.1

This sample script uses the Netscape APIs (Application Programming Interface) for the Macintosh.

```
-- Open URL AppleEvent code
_______________________________
on activateNetscape pageName

  global NetscapeName

  -- open URL apple event
  send pageName to program NetScapeName with "WWW!OURL"  -- openURL Event ID
  put the result into getResult
  if (getResult is 0) or (isnumber(char 1 of getResult)=false) then
     answer "Program Error ! Can't find " & netscapeName
      reset
  else
        put empty into eventString          -- wait for return appleevent
        put 1 into count
     repeat until (eventString is not "No current Apple event.") or (count=100)
        add 1 to count
        request appleEvent data with keyword "PREN"       -- End Session Event ID
        put the result into eventString
   end repeat

   -- if page opened then bring browser to the front, else report error
   if count<100 then
    send "" to program NetscapeName with "WWW!ACTV"  -- Activate Web browser Event ID
    close this window
   else
    answer "Program Error - " & netscapeName & " page not loaded"
   end if
  end if
end activateNetscape
```

References

[1] Fluckiger, François. *"Understanding Networked Multimedia Applications and Technology."* Prentice Hall. 1995: p. 43.

[2] The MIDI Manufactuers Association Technical Standards Board. MIDI 1.0 Detailed Specification. Hollywood, CA: International MIDI Association.

[3] Mooore, Richard. *"Elements of Computer Music."* Prentice Hall. 1990: p. 61.

[4] Fluckiger, François. "Understanding Networked Multimedia Applications and Technology." Prentice Hall, 1995 p. 73

[5] "A Brief Overview of the Proposed General MIDI Level 1 Specification" (http://www.eeb.ele.tue.nl/midi/GMGS.html) by Heini Whitagen

[6] "The Looming Video Game Wars of 1995." *Next Generation* vol. 1, p. 37-48

[7] "QuickTime VR: A New Approach to Virtual Reality." (http://www.info.apple.com/dev/appledirections/jul95/qtvr.html) by David Gleason

[8] Fluckiger, François. "Understanding Networked Multimedia Applications and Technology." Prentice Hall, 1995 p. 397.

[9] Fluckiger, François. "Understanding Networked Multimedia Applications and Technology." Prentice Hall, 1995 p. 332.

[10] Carroll, Jim, and Rich Broadhead. *Canadian Internet Handbook.* (1995 ed.) Prentice Hall Canada Inc., 1995. p.41

[11] Arnett, Matthew Flint, et al. *Inside TCP/IP.* New Riders Publishing, 1994, p.50-70.

[12] Arnett, Matthew Flint, et al. *Inside TCP/IP.* New Riders Publishing, 1994, p. 80-82.

[13] Fluckiger, François. "Understanding Networked Multimedia Applications and Technology." Prentice Hall, 1995 p. 599.

[14] Berners-Lee, T., Cailleau, R., Groff, J., and Pollerman, B. "The World Wide Web, the Information Universe." *Electronic Networking: Research, Applications and Policy*, vol. 2, 1992.

[15] "MIME (Multipurpose Internet Mail Extension) " (http://www.oac.uci.edu/indiv/ehood/MIME/MIME.html)

[16] "The Ethernet Page." (http://www.ots.utexas.edu/ethernet/main.html) by Charles Spurgeon

[17] Boisseau, M, M. Demange, and J.-M. Munier. *High Speed Networks.* trans.John C.C. Nelson, New York: John Wiley & Sons, 1994. p. 43-59.

[18]Brandenburg, Karlheinz, and Gerhard Stoll. "ISO-MPEG-1 Audio: A Geneic Standard for Coding of High-Quality Digital Audio."*JAES* Oct. (1994): 780- 792.

[19] Ozer, Jan. "The CD-ROM Publisher's MEPG Primer." *CD-ROM Professional.* June 1995: p.82

[20] "Welcome to Netscape"
(http://home.netscape.com/)

[21] "HotJava Home Page"
(http://java.sun.com/)

[22] "Interactive CatBand"
(http://www.nasoft.com/catband/index.html)

[23] "The VRML Home Page"
(http://www.ncsa.uiuc.edu/General/VRML/VRMLHome.html)

[24] "Extending WWW to Support Platform Independent Virtual Reality"
(http://vrml.wired.com/concepts/raggett.html)
by David Raggett

[25] "VRML Audio Extension."
(http://hyperreal.com/~wlm/vrsl.html)
by William Martens

[26] "RealAudio"
(http://www.prognet.com/)

[27] "CU-SeeMe Desktop Videoconferencing"
(http://www.indstate.edu/msattler/sci-tech/comp/CU-SeeMe/index.html)

[28] Array Communications of the ICMA, vol. 15, no.2. Summer 1995: p. 12.

[29] "The MBone Information Web"
http://www.best.com/~prince/techinfo/mbone.html)
by Vinay Kumar

[30] "MBone Desktop Application Software"
(http://www.best.com/~prince/techinfo/mc-soft.html)

[31] Nielsen, Ole. "MIDI and Audio via ISDN." *Proc. of the Intl. Computer Music Conference, ICMA, Denmark* (1994): 451-454.

[32] Fober, Dominique. "Real-Time Data Flow on Ethernet and the Software Architecture of MIDIShare." *Proc. of the Intl. Computer Music Conference, ICMA, Denmark* (1994): 447-450.

[33] Pohlman, Ken. *The Principles of Digital Audio.* 2nd Edition. SAMS, 1992: p.321-332.

[34] Steinmetz, Ralf and Klara Nahrstedt. *Multimedia: Computing, Communications and Applications.* Prentice Hall, 1995 p. 189.

[35] Brackenridge, Billy. "One in a Billion: Universalizing the CD Format." *Morph's Outpost* April 1995: p. 18.

[36] ["Enhanced CDs"
(http://www.amp.apple.com/program/imt.html)

[37] Schwerin, Julie. "Video CD and High-Density CD Futures."*CD-ROM Professional.* June 1995: p.36.

[38] Barker, John. "Video CD: The Argument." *CD-ROM Professional.* June 1995: p. 49.

[39] High-Density CD-ROMs and drives to support these discs are predicted by 1996. Schwerin, Julie. "Video CD and High-Density CD Futures."*CD-ROM Professional.* June 1995: p. 37

[40] Parker, Dana. "High-Density and Re-inventing The Disc." *CD-ROM Professional.* June 1995: 20-33.

[41] McGowan, Chris, and Jim McCullaugh. *Entertainment in the Cyber Zone.* New York: Random House, 1995. p.180

[42] Davis, Fred. "I Want My Desktop MTV." *Wired* July 1994: p.68

[43] Gosney, Michael. "Todd Rundgren: Awakening the Interactive Muse." *Keyboard* August 1993: 50-56+.

[44] Potyen, Paul. "Distorted Interactivity: Pop Rocket's Multimedia Vision." *Mix* August 1994: 74.

[45] "CD-Plus Audio CDs"
(http://www.eden.com/cdplus/)

[46] Rule, Greg. "Gingerbread Man, A Case Study: The Making of an Enhanced CD Title." *Interactivity* July/August: 27-33.

[47] "Interactive Games on the Web"
(http://www.cs.cmu.edu/afs/andrew/org/kgb/www/zarf/games.html#games)

[48] "The MUD Resource"
(http://www.cis.upenn.edu/~lwl/mudinfo.html)

[49] "The WAXweb"
(http://bug.village.virginia.edu/)

[50] "Games and Other Diversions"
(http://java.sun.com/applets/alpha/applet/appletSites.html#games)

[51] "Geffen Records"
(http://geffen.com/)

[52] "Sony Music Online"
(http://www.sony.com/Music/MusicIndex.html)

[53] "Atlantic Records"
(http://www.atlantic-records.com/)

[54] "MCA Records"
(http://www.mca.com/mca_records/index.html)

[55] "Netscape API for the Macintosh."
(http://home.netscape.com/newsref/std/mac-remote-control.html#foot3)

[56] "Flypaper"
(http://www.pass.wayne.edu/~eric/flypaper/)

[57] "Firefly: Your Personal Music Recommendation Agent"
(http://www.agents-inc.com/)

[58] "The Autonomous Agents Group"
(http://agents.www.media.mit.edu/groups/agents/)

[59] "Going Hybrid: The Online/CD-ROM Connection." *New Media* June 1995: p.38.