NOTE TO USERS

This reproduction is the best copy available.

UMI
THE COALESCEANCE OF MUSIC AND THE INTERNET: A HYBRID SOLUTION FOR THE USE OF MUSIC MATERIALS IN WORLD-WIDE WEB PUBLICATION

BY

ELBERT H. MCLAUGHLIN

FACULTY OF MUSIC
McGILL UNIVERSITY, MONTREAL
NOVEMBER 1996

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

© ELBERT MCLAUGHLIN, 1996
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-44097-4
The World-Wide Web (WWW) is currently witnessing a dramatic trend toward the incorporation of multimedia elements into on-line publishing. These elements can be very effective for communicating ideas that are difficult to explain with text alone. For the presentation of music, the WWW brings the capability of embedding audio files (recordings) into a document, thus providing audible musical excerpts in addition to textual references to selected points of interest in a musical score. However, there are technical challenges and copyright issues to be addressed when providing musical excerpts on the WWW. The research undertaken in this study brings one possible solution to these issues through the design and implementation of a hybrid system using a combination of the internet and commercially available audio compact discs.
Il y a actuellement une tendance prononcée vers l'incorporation d'éléments multimédia dans les documents publiés sur le World Wide Web (WWW). Plusieurs idées qui sont difficilement exprimées en caractères de texte seulement peuvent être communiquées d'une manière frappante par l'usage de ces éléments multimédia. En ce qui concerne le musique, le WWW offre la possibilité d'intégrer des fichiers audio (enregistrements) dans une page Web, permettant ainsi la présentation d'extraits musicales en plus de l'information textuelle sur les points d'intérêt qui peuvent figurer dans la partition. Cependant, on ne peut négliger les difficultés d'ordre technique et les problèmes de droits d'auteur qui se présentent dans une telle formulation. La présente étude apporte une solution à ces problèmes par le dessein et l'exécution d'un système hybride se servant à la fois du réseau Internet et de disques compacts audio disponibles commercialement.
ACKNOWLEDGMENTS

Throughout the research and writing of this thesis there have been many individuals without whom the end result may never have materialized. First, I would like to thank my thesis advisor, Bruce Pennycook, not only for his support and invaluable advice throughout the study, but also for bringing together such an eclectic mix of students to the program. Their ideas, suggestions, expertise, and enthusiasm have greatly influenced my development and are in no short order responsible for inspiring and motivating my work. To the following I am indebted: David McIntyre, Eli Breder, Sean Terriah, and Kyle Dawkins.

I am, of course, the product of my past, and it would be irresponsible to think that I have made it this far without the help and guidance of all my teachers and mentors. I refrain from referring to them as my 'former' teachers, for their words and teachings are still with me today.

Finally, I am forever grateful for my family's complete support and confidence in my abilities. The time, caring, and resources spent in ensuring that I always had the absolute best is what has made everything possible. And I would especially like to thank Patty for enduring the year that we have been apart, whose love and understanding makes it all worthwhile.
# Table of Contents

1. Introduction ........................................................................................................... 1

2. Current Technologies for Delivering Multimedia Content ............................... 4
   2.1 Preamble ........................................................................................................... 4
   2.2 The World-Wide Web ...................................................................................... 5
   2.3 HyperText Markup Language ......................................................................... 6
   2.4 Embedding Multimedia Content in an HTML Document .............................. 7
   2.5 Interactivity ..................................................................................................... 8
   2.6 Survey of Multimedia Distribution Methods ................................................. 9
   2.7 Conclusion ....................................................................................................... 11

   3.1 Preamble ........................................................................................................ 13
   3.2 Technical Issues .............................................................................................. 14
   3.3 Legal Issues .................................................................................................... 20
   3.4 Conclusion ...................................................................................................... 23

4. Software Design .................................................................................................... 24
   4.1 Forward ........................................................................................................... 24
   4.2 Background of Enabling Technologies .......................................................... 25
   4.3 Design Considerations and Implementation ................................................ 28
   4.4 Other Software Solutions ............................................................................... 33

5. System Trials .......................................................................................................... 35
   5.1 Academic Paper: Analysis of Brahms Op.76 No.4 ....................................... 36
   5.2 Course Supplement: Listening Guide for Music History 101 ....................... 39
   5.3 General Music Education: Children's Interactive "The Nutcracker" ............. 40

6. Conclusion ............................................................................................................... 41

Appendix: User Guide ............................................................................................... 43

Bibliography .............................................................................................................. 53

References .................................................................................................................. 56
Chapter One

Introduction

The internet has proven itself as a new medium for publishing, bringing with it an abundance of new capabilities and possibilities. While most documents are static entities which do not permit interactive responses by the end-user, the World-Wide Web (WWW) provides point-and-click navigation through documents. Many of today's WWW publications incorporate interactive multimedia elements such as animation, video, and audio clips. These elements can be very effective for communicating ideas which are difficult to explain with text alone.

In discussing music it is difficult to communicate effectively without direct access to musical excerpts. For example, musicologists have been obliged to refer to measure numbers and perhaps a graphic reproduction from a musical
score when talking about a particular musical phrase. Presenting musical materials on the WWW brings the capabilities of embedding actual audio files into a document thus providing audible music excerpts. There are, however, several issues which merit discussion concerning music and audio files on the internet.

There is an inherent technical challenge in distributing audio files. File sizes are large because digitized audio data cannot be compressed easily. There are algorithms to compress audio data at the expense of fidelity, but the results are often unacceptable to the musically trained ear. Hence, waiting for audio files to download (that is, transfer the audio data over the internet from a remote file server to the end-user's computer) can be a time consuming and frustrating experience, often proving this potentially powerful communicative tool to be more trouble than its worth.

There are also copyright issues which cannot be ignored. Most music audio files on the WWW are extracted from commercially available CD's and records. By providing these download-able audio files on the internet the author becomes guilty of copyright infringement.

This thesis will demonstrate new software tools developed to overcome the problems outlined above. The solution is a hybrid system of network based information and local media. Segment information is delivered via a server to control playback of an audio CD on the CD-ROM drive in the end-user's computer. A user browsing a WWW page will be able to click on an icon to hear a musical example and the specified audio will be played locally from the user's CD-ROM drive. There are advantages to this system; the response time between clicking and hearing a musical excerpt is nearly instantaneous as there are no audio data being transferred over the internet, and the audio fidelity is that of the
compact disc in the drive. Furthermore, all copyright issues regarding the distribution of audio files disappear because the user is responsible for providing the CD.

The following chapters of this document will describe recent developments in the area of music on the internet, new challenges, and a solution to efficiently, effectively, and legally present music materials on the internet. This document includes a full description of the software developed specifically for this thesis. Chapter Two provides a summary of current technologies for distributing audio and music information over the internet, and discusses how it may be utilized for producing media-rich content for internet publication. Chapter Three discusses the problems, challenges, and implications of delivering audio and music via the internet. Chapter Four provides a detailed description of the work involved in developing the software for segmenting an audio CD and providing control over a local CD-ROM drive via WWW browser and internet server. Chapter Five demonstrates this software in real-world applications with examples ranging from the publication of academic papers to children's general music education. Chapter Six presents the conclusions to the paper. A User Guide to the software appears in the Appendix. The Bibliography lists sources used throughout the research of this thesis as suggestions for further reading. The References section lists sources of specific examples sited in this document.
Chapter Two
Current Technologies for Delivering Multimedia Content

2.1 Preamble

The dominant medium for delivering multimedia content on computers is CD-ROM. However, in recent months the World-Wide Web (WWW) has been extended to incorporate multimedia elements such as audio, video, animation, and enhanced user interactivity, slowly blurring the lines between the two: optical disk and WWW served technology. The proposed Hyper-Text Markup Language (HTML) 3.0 standard accepts author-defined embed tags permitting customized functionality of a WWW publication. (HTML is described in detail in section 2.3). Previously the common practice for including non-HTML media in a WWW page was to refer to an FTP (File Transfer Protocol) site, download the "foreign" file to the user's computer, and use external
"helper-applications" to decode and handle the data. Examples of such non-HTML files include most multimedia content: audio and music files (i.e. AIFF - Audio Interchange File Format, WAV - Microsoft Wave, MIDI - Musical Instrument Digital Interface), video (i.e. MOV - QuickTime Movie, AVI - Microsoft Video), or any other proprietary file format (i.e. PS - Adobe PostScript, or PDF - Adobe Acrobat). Today with the advent of WWW browsers which incorporate open architectures (i.e. Netscape Navigator, Microsoft Internet Explorer), custom "plug-ins" are being developed to provide new functionality to WWW browsers, thereby reducing the need for external helper applications.¹ ² This means that an audio or video clip may reside as an embedded object within a document, much like a graphic illustration in conventional publishing. Compared to juggling multiple applications and windows on a computer screen, in-line plug-in functionality provides a less distracting and more natural way of reading on-line hypermedia material.

2.2 The World-Wide Web

The World-Wide Web (WWW) emerged from the cooperative project initiated in 1989 by CERN, the European Laboratory for Particle Physics.³ The goal of this 19 country International Treaty Organization was to develop communication protocols for linking together large databases of research documents to make them accessible to thousands of scientists geographically spread around the world. The protocols and specifications which resulted from the CERN project became widely adopted by the internet community at large, evolving into a world-wide initiative. There is no central authority governing the WWW as anyone may author and link documents together from any other server on the internet.
2.3 HyperText Markup Language

HyperText may be defined as text with selectable markers, or "links," to other documents. The HyperText Markup Language (HTML) is a language for specifying the layout of text, and methods for linking to other documents. Linked documents may also include images, audio, video, or any other kind of computer file. The language consists of unique "tags" which a WWW browser interprets to format and display the HTML document correctly. (see fig 2.3.1)

![fig 2.3.1 An HTML tag](image)

Text or images may link to other documents through specifying a Uniform Resource Locator (URL). A URL defines the syntax for locating documents on the World-Wide Web. It consists of several parts: the transfer protocol (such as HTTP or FTP), the internet name of the host machine, and the name of the file with its directory. (see fig 2.3.2)

![fig 2.3.2 A Uniform Resource Locator (URL)](image)
When a user clicks on a hypertext link the WWW browser retrieves and displays the requested document, providing a mechanism for navigating through HTML documents. A URL may point to any kind of file. The manner in which data from a requested URL is handled and presented depends on the WWW browser. Guidelines set as a result of the WWW Initiative encourage that all WWW browsers interpret and format HTML "tags" in a manner which ensures a consistent document page-layout regardless of the end-user's computer platform.

2.4 Embedding Multimedia Content in an HTML Document

In the context of the WWW, multimedia shall be defined as the use of any other media in conjunction with text. Such other media may include audio, video, animation, or MIDI. While the HTML standard was originally conceived as a method for linking hyper-text documents together, it also serves as an efficient way of linking non-text documents. As there is no governing body of the WWW, browsers may be designed to support new HTML tags providing enhanced functionality which other browsers may not understand. Fortunately, the guidelines set forth by the WWW Initiative request that browsers simply ignore unrecognized HTML tags as opposed to alerting the user by producing on-screen error messages. An example of a custom tag is Netscape Navigator's "embed" tag. (see fig 2.4)

```html
<embed src='http://www.music.mcgill.ca/fanfare.midi' width=32 height=32 play=true>
```

![Fig 2.4 Example of a Netscape Navigator 'embed' tag](image-url)

Chapter Two - Current Technologies for Delivering Multimedia Content
While other browsers will simply ignore this tag, it will cause Navigator to load a specialized program commonly called a "plug-in," effectively extending the functionality of the original Netscape Navigator software. The appropriate plug-in will be invoked to handle incoming data based on the Multipurpose Internet Mail Extension (MIME) type of the specified URL. (MIME described in detail in section 4.2.2) In the example shown in fig 2.4, the document "fanfare" of type "midi" will be retrieved from the server "www.music.mcgill.ca" and embedded in the calling HTML document with attributes "width," "height," and "play." This mechanism provides virtually limitless possibilities for adding functionality enhancements to an HTML document. For instance, Yamaha MIDI-Plug provides Navigator with a software synthesizer and the functionality for playing MIDI files directly from within a WWW page, effectively adding music to an HTML document.4 (MIDI files described in detail in section 3.2.2)

2.5 Interactivity

Interactivity refers to the real-time response of a system to user commands. To this end, standard HTML documents could be considered interactive, though only to a limited extent. While HTML documents provide the freedom to link one document to the next, all information remains static until the user's next mouse click.

In an effort to provide greater interactivity, Macromedia Corporation developed the Shockwave plug-in.5 This plug-in provides a WWW browser with the capability to embed Macromedia Director files within HTML documents. Director is an industry standard interactive multimedia authoring...
tool commonly used to create CD-ROM titles. It provides a framework and scripting language (Lingo) for controlling and manipulating audio, video, and animation, in conjunction with real-time user interaction. Because *Director* is designed to respond to complex user interaction, a WWW browser with the *Shockwave* plug-in installed acquires these capabilities for enhanced on-line interactivity. For example, rolling the mouse cursor over a *Shockwave* graphic may animate an object or play a sound, giving immediate feedback to the user.

2.6 Survey of Multimedia Distribution Methods

There are at present three methods of distributing multimedia content. The data may be shipped entirely on physical media such as CD-ROM, downloaded "as-necessary" over a network such as the internet, or obtained via a hybrid system combining both of these methods.

2.6.1 Local Media: CD-ROM

CD-ROM provides an inexpensive medium for distributing the large amounts of data required by multimedia applications. Due to the success of the audio CD upon which CD-ROM technology is based, the infrastructure for mass producing compact disc media had already been established at the time of CD-ROM's introduction. The media itself is defined to be "platform-independent" (meaning, will play correctly regardless of computer type or manufacturer model) but may be formatted to suite the needs of a specific file system. For example, the ISO9660 standard is widely used on UNIX and PC systems while the Hierarchical File Structure (HFS) is prevalent on the Macintosh platform. The ability to freely access large amounts of data from a single disc quickly made it a prime choice for early interactive multimedia applications such as Cyan's *Myst*.
Grollier's *Multimedia Encyclopedia*, and Peter Gabriel's *Xplora*. While the CD-ROM allows for large quantities of data to be shipped inexpensively, it is by definition *read-only*, making it a closed system. Once the disc is "pressed," the data cannot be altered or updated.

2.6.2 Distributed Media: Internet

The internet permits unlimited transfers of multimedia content. The essential difference compared to CD-ROM is that the content does not reside locally with the user. All information must be transferred via internet servers as requested by software such as a WWW browser. Plug-ins for WWW browsers such as Apple Computer's *QuickTime Plug-In* (video, QuickTimeVR), Macromedia's *Shockwave Plug-In* (interactivity, animation), and Progressive Networks' *RealAudio Player* (streamed audio in real time) has made possible such "web-adventures" as Hollywood Pictures' *Mission Impossible* site, and Paramount Pictures' *Independence Day* site.\(^7\)\(^8\)\(^9\) While delivering multimedia content via the internet allows for new and updated information, internet data transfer rates are still only a fraction of the data throughput attainable from a CD-ROM. This significantly restricts the practical file sizes of deliverable multimedia content.

2.6.3 Hybrid Systems: Local and Networked Information

While it is expected that internet bandwidth will eventually match or even exceed data throughput rates of CD-ROM, a hybrid system which combines fixed data on CD-ROM and dynamic data from the WWW may be the ideal solution for bringing the user the "best of both worlds." By distributing the bulk of the multimedia data on CD-ROM and relying on the internet only for new
and updated information, each medium may be exploited for what it does best. (see fig 2.6.3)

Such a system is capable of providing rich multimedia content distributable only on CD-ROM with a pipeline to new material from the WWW; a scenario not possible using either medium alone. The hybrid system may be invoked from a CD-ROM which seeks out information from the internet as necessary, or the reverse where a WWW browser seeks large multimedia files from a local CD-ROM. While such a system has yet to be used in many commercial applications, it is only logical to assume that this is the next evolutionary step in interactive multimedia design.

2.7 Conclusion

It has been established that CD technology remains the most effective method of distributing high volumes of data. The WWW is widely accessible and provides a means for delivering dynamic and updated information, but
internet bandwidth constraints significantly limit the speed at which this data may be transferred. A combination of these two distribution channels brings together the "best of both worlds" in a hybrid system which allows large amounts of data and dynamic information.
Chapter Three
Issues of Delivering Music and Audio in WWW Publication

3.1 Preamble

There are several issues which should concern any HTML author who is considering the use of audio or music media in WWW publication. Of concern are issues regarding the target audience's hardware requirements; will the target user viewing a document have the necessary computer equipment to take advantage of the audio content? Also of concern are the rights of creative ownership and copyright laws when distributing protected materials. Surprisingly, often both of these issues are overlooked by many WWW authors.
3.2 Technical Issues

The two most prevalent methods for delivering music over the internet are audio files and MIDI files. Audio files contain digitized sound data (usually recorded from a conventional audio source such as compact disc or cassette) which is converted during playback into an analog signal. MIDI files contain note data (created using some form of MIDI sequencer software) which is reconstructed and synthesized during playback. There are advantages and disadvantages to using each file type.

3.2.1 Audio Files

An audio file is a digital representation of audible sound. To convert an analog sound source, the analog signal is fed through an Analog-to-Digital Converter (ADC) which converts the incoming signal into a digital stream. (see fig 3.2.1a)

![fig 3.2.1a Analog to Digital Conversion](image)

The most common type of digital encoding is Pulse Code Modulation (PCM). PCM is the encoding method used for CD audio, AIFF, and WAV files, the most common formats for Apple and Wintel type computers. There are two important factors which affect PCM sound quality: sample rate and bit-depth. (fig 3.2.1b) The sampling rate is the frequency at which the amplitude of the audio
signal is measured. The bit-depth determines the number of discrete steps available to measure the amplitude between silence and maximum volume.

![Diagram showing bit-depth and sampling rate](image)

fig 3.2.1b Linear PCM encoding

For all forms of PCM encoding, the audio frequency response is dependent on the sampling rate. The Nyquist Theorem states that the maximum frequency range is slightly less than half the sampling rate. To cover the frequency range of human hearing, which spans from 20Hz to 20KHz, a sampling rate greater than 40KHz is necessary. Hence the sampling rate of all audio CDs was set at 44.1KHz and professional recording equipment was set at 48KHz. Both exceed the minimum Nyquist frequency for full audio band reproduction. Different computer platforms use their own audio file formats: SDII and Sys7 formats are used primarily on the Macintosh, WAV format is commonly used on Windows machines, and AIFF format is used on SGI. Conversion between any of these file types retains all digital information and hence reproduces identical sound during playback.

During playback the analog-to-digital conversion process is reversed by passing the digital stream through a Digital-to-Analog Converter (DAC). (fig 3.2.1c)
The resulting analog signal may then be amplified through a conventional speaker system or headphones.

3.2.2 MIDI Files

MIDI files do not contain any audio information. Rather, a MIDI file contains series of note data, or performance data, which a computer or synthesizer must reconstruct. (see fig 3.2.2) A good analogy would be to compare the file to a piano-roll for player pianos. A MIDI file contains information such as when a note is played, how loud the note is played, what instrument (program number) should play the note, where in the stereo field the note should be panned, and other performance characteristics such as pitch bend and modulation.
In September 1991 the MIDI Manufacturers Association (MMA) and the Japan MIDI Standards Committee (JMSC) agreed on a specification for providing a "minimum level of performance compatibility among MIDI instruments" called the General MIDI System Level 1 (GM) specification. Any synthesizer which adheres to the GM specification contains a required set of 128 sounds assigned to designated program numbers, and must respond to a minimum

---

Chapter Three - Issues of Delivering Music and Audio in WWW Publication
number of performance control data. The purpose of this agreement was so that any MIDI file created using a GM synthesizer can be played back on another GM synthesizer with the proper instrument sounds, regardless of the manufacturer. For the purpose of distributing MIDI over the internet, GM is an invaluable specification allowing, for example, a MIDI file created in Europe on an Amiga computer with an external GM sound module to play back properly on a PC using an internal sound card in North America. The sound quality during MIDI playback is entirely dependent on the quality of the playback synthesizer. While some professional synthesizers can reproduce rather convincing performances, it is safe to assume that the majority of internet users will not have such high-quality MIDI playback facilities.

3.2.3 File Size versus Limited Bandwidth

While high-quality audio files might be the preferred choice of musicians over synthesized note data, the current realities of internet bandwidth limitations make waiting for file transfer unrealistic. To download a 10 second CD-quality audio excerpt via a 28.8 Kbps modem, which is used by the majority of the internet community, takes over 8 minutes! With a theoretical maximum transfer rate of 3 KBytes per second via 28.8 Kbps modem, downloading 10 seconds of full bandwidth audio (16-bit resolution, 44.1 KHz sampling rate, stereo) at 150 KBytes per second would take: 10 seconds \times 150 \text{ KB per second} / 3 \text{ K per second} = 500 \text{ seconds} = 8 \text{ minutes 20 seconds}. Clearly an alternative is necessary to make audio delivery over the WWW practical. One solution is to drop the audio resolution to decrease file size. For example, simply summing a stereo file to mono and eliminating one channel cuts the file size in half. By decreasing the sampling rate and bit depth the file size can be further
diminished. Logarithmic PCM encoding at 8-bits/sample, as used widely in digital telephony (usually referred to as 8-bit μ-law), is capable of delivering sound quality nearly equal in quality to a 14-bit linear encoding, but for some reason this is not widely used. The problem with all of these schemes is that they are a compromise at the expense of audio fidelity.

While many forms of data can be compressed (that is, the data may be modified in its encoding to reduce the capacity to store it and the bit rate required to transfer it), digital audio is the exception. Most compression algorithms rely on redundancies in the data, but audio data lacks such redundancies rendering such algorithms useless. There are algorithms designed for compressing specific audio content, accomplishing data compression by "throwing away" non-essential elements of the sound to achieve smaller file size. For example, one of the widely used internet techniques called RealAudio manages to compress audio data 28:1 resulting in a throughput rate of under 14.4 Kbps.\textsuperscript{11} This is accomplished, however, by eliminating virtually all the audio spectrum data which is not essential for the preservation of speech intelligibility. RealAudio achieves significant audio data compression, but the resulting sound quality is worse than AM radio, making it undesirable for any application other than spoken voice.

To minimize or even eliminate the time spent waiting for file transfers, a technology exists called streaming which permits the user to hear audio while it is being transferred. Examples of streaming technology include AudioActive's AudioActive Player and Apple Computer's QuickTime Plug-In.\textsuperscript{12} Both of these technologies allow the end user to hear or view, respectively, incoming data as it arrives from the internet. While this is a vast improvement over waiting the full duration of time it takes to download a file, it still places a great burden on
internet bandwidth to sustain the minimum transfer rates necessary for seamless playback.

An alternative to audio files is to use MIDI files. A typical 10 second excerpt might contain as many as 200 notes, but the resulting file would still be less than 2 KB in size and would take less than one second to download via a typical 28.8 Kbps modem. At 8 bytes per note, the total file size would be: 8 Bytes * 200 notes = 1600 Bytes = 1.6 KB.

3.3 Legal Issues

Technology is making it increasingly easy to combine media created by others to produce "new" multimedia presentations. The result of this has been the proliferation of WWW pages full of media components. Unfortunately, if this media is being presented without the permission of copyright owners, the author of a page becomes guilty of infringement. This is a serious issue concerning music used in multimedia presentations given that there is a large quantity of audio and MIDI files made available on the WWW which are not in the public domain.

Copyright protection on the internet is somewhat problematic in that the internet knows of no geographic or national boundaries. Intellectual property laws vary from country to country and the protection of rights cannot easily be enforced beyond domestic borders. Discussion herein will focus on United States copyright law since the majority of internet users fall under its jurisdiction. (International copyright issues are beyond the scope of this document).
Copyright Protection

Copyright is designed to help encourage creative output by providing an infrastructure to protect the hard labor of individuals who produce new works. Copyright protection automatically arises once an original work is "fixed" in a tangible medium by an author. (Filing for protection is only necessary before filing an infringement suit). Works protected by law that are of interest to this discussion include musical works and sound recordings. A copyright owner is entitled to the following exclusive rights over his/her music: reproduction, modification, and distribution. Note that public performance and public display rights which protect literature, dance, drama, and visual arts do not necessarily apply to music sound recordings. The reproduction right preserves the right to copy, duplicate, transcribe, or imitate the work. The modification right allows the author to create derivative works based on the original. Distribution rights includes the authority to freely distribute copies of the work by sale, rental, lease, or lending. A person found violating any of these rights is infringing. When dealing with sound recordings several layers of caution must be observed. The composer, performer, and recording label are each likely to own specific rights pertaining to the recording. When a WWW author chooses to include music in a site, licensing agreements may be required to be established with each and every party involved.

Exceptions

It is not necessary to obtain a license to use musical material if the use is "fair use" or if the music is in the public domain. (Note that there is no "fair use" clause in Canadian copyright law, leaving only works in the public domain to be used at will). Fair use must be determined on a case-by-case basis by
considering the purpose of use, the nature of the copyrighted work, the amount or substantiality of the work used, and the value of the protected work. Also, works in the public domain may be used free of licensing. It is the responsibility of the author to confirm that a work is indeed public. Note that as of March 1, 1989 a work may be fully protected without requiring an attached copyright notice.

For most applications of music on the WWW, the fair use clause is not likely to exempt licensing arrangements. Any multimedia WWW site which uses music to enhance a presentation is implicitly acknowledging the value of the music and such use of non-licensed material would be deemed as "unfair use."

3.3.3 MIDI files and copyright

Until very recently, MIDI files provided a solution for license-free music on the WWW because they were not protected by copyright laws. Because copyright laws have been traditionally designed around material property, a MIDI file was deemed exempt as it was considered an "intangible" item. However, the current views of the MMA and the U.S. Copyright Office equate MIDI files with CDs and audiocassettes by focusing on the intellectual content, rather than the "property" or physical medium itself. This means that MIDI files of public domain works may now fall under protection if the file contains a unique and identifiable performance. A performer is still free to create and distribute a MIDI file of music in the public domain, but others wishing to use this file in their own work may need to reach licensing agreements as the performance rights are reserved for the original creator.
3.4 Conclusion

A hybrid system of internet and local CD provides a solution for not only the technical challenges of controlling audio in real time, but also for honoring copyright laws when using protected materials. Even when the internet attains data transmission speeds capable of delivering full-bandwidth CD-quality audio to all, the legal issues will still remain an important concern.
Chapter Four
Software Design

4.1 Forward

The solution to the technical challenges of controlling audio in real-time and the issues of honoring copyright laws is a hybrid system of delivering data from both the internet and local CD-ROM. In order to provide this functionality a plug-in must be developed with the capabilities to i) communicate with the local CD-ROM drive, ii) interpret a file containing timing information for specific CDs, and iii) provide an intuitive user interface for direct control of the plug-in from within a WWW document. The segment file which the plug-in needs for timing information requires an external application to be built with the capability of i) defining bar lines, ii) defining musical regions (based on bar lines), and iii) consolidate timing information for multiple CDs of the same musical
work. The software described below was developed by the author to realize this hybrid system solution.¹³

All software was developed on the Macintosh platform using the Metrowerks' CodeWarrior Integrated Development Environment in C/C++. In addition to the standard libraries for MacOS development, the Netscape Plug-In API (version 2.0) was used to implement the plug-in component.

4.2 Background of Enabling Technologies

Before embarking upon the details of software design, an explanation of the building blocks is in order. As the software must be capable of seeking music material with the greatest possible accuracy, it is necessary to understand how audio data is stored on the compact disc. It is also necessary to understand the client-server model which allows the plug-in to receive information regarding musical segmentation from an HTML page.

4.2.1 The "Red Book" Standard (CD-DA) Data Encoding

All data on a CD-DA is arranged in frames of 291 bits in length. (fig 4.2.1)¹⁴

```
<table>
<thead>
<tr>
<th>Sync</th>
<th>Subcode</th>
<th>Audio data</th>
<th>Parity</th>
<th>Audio data</th>
<th>Parity</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 bit</td>
<td>8 bit</td>
<td>96 bit</td>
<td>32 bit</td>
<td>96 bit</td>
<td>32 bit</td>
</tr>
</tbody>
</table>
```

(291 bits)

fig 4.2.1 Breakdown of a CD-DA frame

Therefore, the smallest recognizable increment on a CD-DA is the frame. The subcode region is where timing and TOC (table of contents) information is

---

Chapter Four - Software Design 25
stored. The audio data chunk of 96 bits appears twice per frame to allow for stereo: 96 bits for the left channel, 96 bits for the right channel. A little math provides the following statistics. At 16-bit resolution: 96/16 bits = 6 samples per frame. At a sampling rate of 44100 Hz: 44100/6 samples = 7350 frames per second. However, due to the way in which subcode information is interleaved for error correction, a CD-DA cannot be accessed at 1/7350 second precision. The minimum amount of information which may be retrieved at one time is a subcode block. This block is 784 bits in length. With 8 bits of subcode data per audio frame, this requires 784/8 bits = 98 frames to complete a single subcode block. Because it is only possible to access a specific location based on subcode data, the smallest seekable increment is thus 7350/98 frames = 75Hz, or 1/75 of a second. Hence, the syntax for addressing a specific location on the disc is by minute: second: frame. It is stated in the "Red Book" that the maximum number of frames per second is 75, the maximum number of seconds per minute is 60, and the maximum number of minutes per disc is 74.

There are several different CD formats, each designed for specific applications. Of special interest to this thesis is the CD+G (sometimes referred to simply as "CD+") format. The "+G" stands for graphics, though in truth any form of data may occupy this space. This format extends the original storage capacity of the CD-DA format by using the bits if the subcode region which were left empty in the original specification for future use. There are 6 unused bits for every frame. This amounts to roughly 20 million bytes (MB) of potential storage space which is discarded on normal audio CDs. The total subcode region storage space (theoretical maximum) may be calculated as:
Bits per second = 6 bits x 7350 frames per second = 44100 bits
Seconds per disc = 74 minutes per disc x 60 second = 4440 seconds
Total bits per disc = 44100 b.p.s. x 4440 s.p.d. = 195804000 bits
Megabytes per disc = 195804000 bits / 8 bits per byte / 1,000,000 bytes
= 24 MB.

The actual amount of subcode space available is directly proportional to the total length of audio data on the CD. This is because the subcode region only exists as extra space inside an audio frame. If there are fewer audio blocks, there is less subcode data space.

The advantage to using the CD+G format for extended data is that a conventional audio CD player will simply ignore the subcode region used to hold this additional data, making it completely compatible with conventional playback technology. This extra data may be added to what is essentially an audio CD and remain playable on all CD audio equipment.

4.2.2 HTTP and MIME explained

The majority of file transfers on the WWW take place using the HyperText Transfer Protocol (HTTP). The HTTP protocol is a client-server protocol designed for efficiently exchanging information. Its efficiency comes from the fact that it is a stateless protocol; a connection is not maintained between the browser and the server across successive file transfers. Each transfer is viewed as an independent file transfer request, freeing the server's resources to handle other clients.

Before an HTTP file transfer takes place, the client (browser) and server exchange information regarding which file formats are mutually recognized. Multipurpose Internet Mail Exchange (MIME) types are part of this exchange.
MIME is an internet multimedia standard which was adopted in 1992 to support multimedia mail; e-mail with embedded multimedia content. This mechanism allows a mail browser to identify and handle non-text media based upon a three or four letter suffix such as "filename.mime" where ".mime" indicates how to interpret the multimedia data. For example, the suffix ".mov" commonly indicates that the incoming data is of type "video/quicktime." In the case of software developed for this thesis, the suffix ".aucd" indicates the MIME type "audio/x-audiocd" which in turn invokes the plug-in software to be loaded within Netscape Navigator, thus enabling control over an audio CD.

4.3 Design Considerations and Implementation

To bridge communication between a WWW document and local CD-ROM drive, a plug-in must be developed to provide the WWW browser with CD audio playback control. This plug-in will rely on specific timing information delivered via the internet for each CD. Software tools must therefore be developed to segment the CD and export this timing information in a properly formatted data file.

4.3.1 Segment the Audio CD

The first step in preparing an HTML document with embedded audio CD control is to create a segment file with timing information. In order to call a musical excerpt by measure numbers the CD must be segmented by bar lines with references to exact timings. Due to differences in musical interpretation, tempos are certain to fluctuate from performance to performance. It was determined that the most convenient way to mark bar lines is to identify them in real-time while the music is playing. The CDsegmentor application marks a bar
line whenever a user hits the "return" key during playback. The application saves a file to disk containing the exact timings for each bar line, accurate to within 1/75th of a second. (fig 4.3.1)

This segment file may later be edited, if so desired, with the CDeditor application.

4.3.2 Editing the Segment File

The CDeditor application was developed to allow a user to modify an existing segment file. A user may wish to view a file to verify the accuracy of the entered bar lines, or correct any mistakes incurred during the segmenting process. The editor also provides the capability to specify musical regions such as "Exposition" or "Development" by providing measure numbers for the beginning and ending of each region. In the case where there are multiple performances (from different CD's) of the same piece, the editor will import other segment files and append them to create a multi-disc segment file. (see fig 4.3.2)
The application is always aware of which CD is currently in the CD-ROM drive and will know where the exact location of a specified measure number resides on that particular disc. CD's are identified by the total number of physical blocks on the disc. The editor allows the user to modify and edit any information regarding the location of bar lines or measure numbers, region names, region location, and disc information such as title, composer, performer, and recording label. This file is then saved with the suffix ".aucd" and is stored on an HTTP server. The AudioCD Plug-In for Netscape Navigator will access this file by calling its URL.

4.3.3 The In-line Plug-in

The function of the AudioCD Plug-In is to provide a mechanism which gives the WWW browser full audio playback control over the user's CD-ROM
drive. Plug-ins are invoked through the use of embed tags. (see fig 4.3.3a) Attributes within an embed tag inform the WWW browser of which plug-in should handle the data (based on MIME type), and may include parameters specifying how to handle the data. The AudioCD Plug-In's routines are therefore called if the SRC (stands for source file) attribute points to a file with the suffix ".aucd" or the TYPE attribute is set to "audio/x-audiocd."

```
<EMBED TYPE="audio/x-audiocd"
WIDTH=144 HEIGHT=24 STARTTIME=94200 STOPTIME=100615>

WIDTH=144 HEIGHT=24 BEGIN=1 END=4>
```

fig 4.3.3a Sample EMBED tags

In the first embed tag in fig 4.3.3a, the TYPE attribute causes an instance of the plug-in to be created, and displays the plug-in's user interface embedded in the WWW page with a WIDTH of 144 pixels and a HEIGHT of 24 pixels. The STARTTIME and STOPTIME attributes specify where on the CD to start playing (94200 = 9 minutes, 42 seconds and 0 frames) and where to stop playing (100615 = 10 minutes, 6 seconds and 15 frames) respectively. This is the simplest method of specifying an audio excerpt, but is also the least "intelligent." The plug-in is given no knowledge about the CD in the drive, and will thus play the specified time markings of any disc which happens to be present. However, for documents which target a single specific CD, this tag may be sufficient as it provides the fastest response time between a play request and the execution. This is because the start and stop times are contained in the embed tag of the
parent HTML document; the additional segment file containing timing information need not be downloaded.

In the second example in fig 4.3.3a, the SRC attribute is passed a URL with the suffix ".aucd" which causes an instance of the plug-in to be created. Again, this displays the plug-in's interface on the WWW page with a WIDTH of 144 pixels and a HEIGHT of 24 pixels. The SRC file is a segment file containing timing information about one or more CD's. The BEGIN and END attributes specify which measure numbers to play, but the plug-in has some work to do to determine how to interpret the arguments. The plug-in must first check the local CD-ROM drive and identify the audio CD by determining the total number of physical blocks. The plug-in then requests a data stream of the SRC URL. (fig 4.3.3b) There must be a corresponding CD in the segment file to ensure correct playback of the requested passage. The plug-in parses through the file to find the exact timings of measures 1 and 3 for the current disc. (The user is notified if the drive is empty or the current CD is not supported in the segment file).

![Diagram of data flow](image)

**fig 4.3.3b** AudioCD Plug-in command/data flow...
An excerpt may also be requested by musical region. The plug-in scans the segment file to find the measure boundaries and parses for timings as described above. Referring to CD location by musically relevant terminology empowers the WWW-author with the ability to focus on the musical ideas rather than the enabling technologies. (For a complete listing of EMBED attributes, see the Appendix: User's Guide).

4.4 Other Software Solutions

The only other tool currently available for controlling a local audio CD via the WWW is Voyager's CD Link application. This program runs as a helper-application, called by the WWW browser to handle URL's with the suffix "vcd" and the MIME type "application/x-cdlink." A helper-application runs as a separate program outside the WWW browser to handle functionality which the browser cannot. A text file containing a start and stop time or track number is downloaded to the client's computer when a URL pointing to a CD-Link document is clicked. The file is automatically passed to the CD-Link application and the CD begins playback. An advantage to using a helper-application instead of a plug-in is that the one time development of the helper-application works with every WWW-browser. A separate plug-in need not be developed for each and every WWW-browser on the market.

The design concept for the CD-Link application does not take into consideration musical concerns. Start and stop times are specified with exact timing information (minute:second:frame) or by track number, not by anything that supports meaningful music structures. Also, there is no support for multiple discs as the application is unaware of the CD in the drive beyond identifying it as being of type CD-DA format.
Voyager does not supply any tools for finding entry and exit points for musical excerpts. However, a *HyperCard* stack by Malcolm Humes and Alexander Rubli, aptly titled *VCD Sequence Generator*, provides a useful graphic interface for generating *CD-Link* files. A feature of the *CD-Link* application which the thesis software does not have is the ability to string together "play-lists" of musical segments. This allows a single click on a URL link to automatically jump through multiple segments on the CD in sequence. Perhaps future versions of the thesis software will be implemented to read and execute *CD-Link* files (.vcd) in addition to its own segment file (.aucd) format.
Chapter Five

System Trials

The thesis software was implemented and tested on an Apple Power
Macintosh 8500/150 (PowerPC 604 microprocessor at 150MHz) and a Macintosh
Quadra 800 (Motorola 68040 microprocessor at 33MHz plus a PowerPC 601
microprocessor at 66MHz). Software used to design these examples include
Finale 3.0 (Coda) for music notation, Photoshop 3.0 (Adobe) for graphic design,
Director 5.0 (Macromedia) to build the Shockwave components, and of course
the thesis software for segmenting and embedding the audio examples. The
demonstration pages have been tested on the above mentioned machines as well
as a Macintosh LC575 (Motorola 68LC40 microprocessor at 33MHz) running
Netscape Navigator v3.0 and the Macromedia Shockwave plug-in.
The following examples are designed to demonstrate various applications of the thesis software and the impact that it can provide for internet publication. The first example is an academic paper with embedded musical excerpts. The objective is to demonstrate the ability to use different performances (from different CDs) for the same analysis. The second example is a course supplement listening guide for a music history class. The CD chosen for this example is a compilation of works from many historical periods. Because it is likely to be in the library collection of most music institutions, it demonstrates a single on-line listening guide which could be used by hundreds of institutions world-wide. The third example is aimed at delivering multimedia "edutainment" on the WWW for general music education. Tchaikovsky's *Nutcracker Suite* was selected to demonstrate the ability to deliver an on-line multimedia storybook.

5.1 Academic Paper: Analysis of Brahms Op.76 No.4

The original idea for the *AudioCD Plug-In* was a result of a discussion among fellow colleagues concerning the problems of presenting music theory analysis on the WWW. The first prototype of the *AudioCD Plug-In* was developed and tested on an analysis of Brahms' *Klavierstücke Op.76 No.4*, written by Shireen Maluf. Maluf's work was originally designed as part of *The Music Library of the Future* project; a McGill University research initiative to develop tools for delivering audio and MIDI files over the WWW, funded by the Canadian Network for the Advancement of Research, Industry and Education (CANARIE).

In this example, MIDI and audio files from the CANARIE demonstrations have been replaced by instances of the *AudioCD Plug-In*. (see fig 5.1) Also, two
of the original graphics have been replaced by interactive animations using Macromedia's *Shockwave* plug-in.

**2.2 Section A**

(a) Harmonic Structure

Op. 76, no. 4 opens with a dominant seventh chord on F which lasts for three measures (Example 1(a)). Naturally, this V7 must resolve to I as shown in Example 1(b), in order to establish the tonality of B-flat major. Or it might move up to V1, the normal substitution for I, as shown in Example 1(c), in order to create a deceptive resolution. But the V7 proceeds with neither of these alternatives and moves instead to IV6 in m. 4 as shown in Example 1(d).

Although the IV6 sounds like a momentary resolution, it does not resolve the seventh of V7. The progression V7-IV6 produces the so-called "stationary resolution of the seventh," which, in fact, is not a resolution at all but a deferral of resolution: the seventh is retained in the IV6. Normally, this results in a prolongation of the seventh through a broader dominant harmony as shown in Example 2: the IV6 functions locally as a pseudo-resolution of the opening dominant but on a slightly larger scale (mm. 1-4). IV6 is a passing chord within a prolonged dominant harmony.

![Example 1](image)

**fig 5.1** The opening of Brahms Op.76 No.4 as displayed in Netscape Navigator

---

Chapter Five - System Trials 37
This demonstration supports three CDs:


The correct passages will play from any of these discs.
5.2 Course Supplement: Listening Guide for Music History 101

In any music history class there is a listening component. Finding an exact location of a referred excerpt on a CD can be taxing and distracting to one's concentration. It would be ideal to request the playback of a passage by referring to its name, rather than constantly referring to the CD liner notes and scanning through multiple tracks and indices.

This is a demonstration of a WWW page designed to assist students studying for an introductory level music history course. (see fig 5.2) Holding the mouse down in the display window produces a pop-up menu containing the musical examples. Releasing the mouse on the desired item will cause the playback of the specified musical excerpt.

The plug-in expects to find the CD:

5.3 General Music Education: Children's Interactive "The Nutcracker"

While originally conceived of for academic use, the AudioCD Plug-In can also assist in the delivery of multimedia entertainment. This is an example of an on-line multimedia story book based on Tchaikovsky's Nutcracker Suite, aimed at a younger audience. (see fig 5.3) The ballet suite was chosen because the programmatic nature of the work lends itself well to visual accompaniments and exciting interactive elements.

The CD used in this demonstration is:

Chapter Six
Conclusions

This thesis has described the issues surrounding the delivery of audio and music content in WWW publications. It has further demonstrated that current mechanisms for incorporating music content in WWW documents are far from practical given the state of internet bandwidth and the average internet user's computer setup. To improve the situation, technical solutions must come either from improving the hardware infrastructure between the client and server machines, or by building new software solutions which address the realities of current internet bandwidth. Regardless of technical challenges, the distribution of music and audio files via the WWW raises serious concerns about protecting copyrighted materials.
The software developed for this thesis is an example of a hybrid solution between networked and local media. It addresses, specifically, both the current realities of internet bandwidth as well as protecting copyrighted material. The software goes beyond this to provide tools for segmenting and previewing audio CDs in a way which is meaningful to musicians. The ability to embed musical examples in a WWW document and randomly access any portion of a musical work is extremely powerful. It opens up new possibilities for designing WWW sites with rich audio content, from on-line music analysis to multimedia entertainment.
Appendix

User Guide

Initial Requirements

In order to use the CDsegmentor, CDEditor, and AudioCD Plug-In software, the following requirements must be met:

(i) A Macintosh computer, preferably 68040 or PowerMacintosh series.
(ii) An Apple CD-ROM drive or equivalent.
(iii) 8-bit (256 color) capable color monitor.
(iv) Netscape Navigator v3.0
(v) Enough RAM to allocate an additional megabyte to Netscape Navigator.

Before a CD may be used in an HTML document, a segment file must be created which holds specific timing information for a selected CD or group of CDs.
The CDsegmentor

The CDsegmentor is a skeletal application designed only to indicate bar lines in real time as a CD is playing. The interface has four buttons: advance track "+", previous track "-", "add measure," and "done."

The CD is always playing while this window is visible. Each time the user clicks on "add measure" (same as hitting the "return" key) the measure number is incremented and the min:sec:frames fields are updated and stored. Clicking on "done" stops the CD and prompts the user with the standard "save file" dialog box. The saved segment file may then be imported into the CDeditor to add features such as region names and multiple CD support.

The CDeditor

The CDeditor provides functionality for defining musical regions and creating multi-CD segment files. Upon opening or importing a segment file from the File menu, a graphical representation of the file is displayed with measure markers above a timeline. These markers may be dragged to the left or right with the mouse for "scrubbing" to the exact location of a measure. The CD will automatically play the new location when the mouse button is released.
**Note:** The CD in the drive must match the CD displayed in the editor in order to move markers around. Either click in the "current CD" window to immediately display the appropriate disc, or select it manually from the View menu.

- **Getting Around in the Editor**

  Maneuvering around the file may be done in several ways. The arrows directly beneath the editor window scroll through the document in 1 second increments. Alternatively, the six buttons in the 'Editor control' panel allow for jumping directly to specific locations.
**Measure**: jump to a specific measure marker.

![Jump to measure number: 67](image)

**Region**: jump to the beginning or end marker of a specified musical region.

![Jump to region marker: piano entrance](image)

**Time**: jump to an absolute time (total elapsed time) on the disc by minutes:seconds.

![Jump to absolute time](image)

**Top**: jumps directly to measure 1.

**Bottom**: jumps directly to the last measure in the segment file.

**Chase CD**: jumps to wherever the CD is currently playing.

*Note: These controls also appear in the Marker menu with $8$-key equivalents.*
Defining Musical Regions

To add a musical region, select "Define New Region..." from the "Marker" menu. The following dialog box will appear prompting for the name, beginning, and ending measure number of the region.

![Dialog box for defining new region]

After clicking "OK," the new region will appear under the timeline in the editor window. Like the measure markers, these markers may also be dragged around with the mouse. This is particularly useful for defining regions which do not start exactly on the bar line.

*Note: a region marker is defined relative to its parent measure markers... if a beginning or end measure marker is moved, the region markers below will shift with them!*

To modify or delete regions, select the appropriate items from the Region menu.

Inserting and Deleting Measure Markers

Measure markers may also be inserted or deleted manually. To insert a measure marker, click on the timeline while holding down the "option" key. To delete a measure marker, click on the marker while holding down the "option" key. Measure markers may also be inserted/added "on-the-fly." This is done by
pressing "8-spacebar" while a CD is playing, much like adding bar lines in real time with the CDsegmentor application.

**Adding New Discs / Creating Multi-CD files**

There are two ways to include multiple CDs in a segment file. One way is to "import" a segment file. This is done by selecting **Import** from the **File** menu. The segment information for imported discs will be appended to the currently open file, and may be selected for display at the bottom of the **View** menu. The other way to add discs is by selecting **New** from the **Disc** menu. This will prompt you for disc information and append the data to the **View** menu. This new disc may then be segmented manually by option-clicking in the timeline, or in real time by playing the desired area and pressing **8-spacebar** on every bar line.

*Note: previously defined region information is not imported. The regions defined in the current document will be used with the imported measure markers and timings. When segmenting a new disc, region markers will automatically "pop-up" as the bar lines are marked.*

**The AudioCD Plug-In**

The **AudioCD Plug-In** is what displays the graphical user interface a reader sees when viewing an **AudioCD Plug-In** enhanced HTML document in Netscape **Navigator**. The plug-in must reside inside the "Plug-Ins" folder, which may be found in the same folder as Netscape **Navigator** itself.

The plug-in should automatically register itself upon launching **Navigator**. However, in the case that it does not a "broken plug-in" icon will appear in place of the **AudioCD** interface. This problem can be resolved by
configuring "helper applications" in the General Preferences... item found in the Options menu.

Be sure to confirm that Navigator will handle the plug-in as expected by selecting the "AudioCD Thesis Plug-In" and clicking Edit.

*Embedding the Plug-In into an HTML Document*

An instance of the plug-in is called whenever an AudioCD Plug-In <embed> tag appears in an HTML document. The basic format of this tag should look like:

```html
<EMBED SRC="segmentfile.aucd" ARGUMENTS=... WIDTH="160" HEIGHT="40"/>
```
Possible arguments include:

**SRC**
The URL of the segment file must be specified. The file must have the .aucd suffix. SRC or TYPE argument is mandatory.
SRC="http://www.music.mcgill.ca/myfile.aucd"

**WIDTH**
This value must change based on the interface used. In the case of the standard user interface (UI), the value should be set to 160. In the case of the small UI, the value should be set to 140. This argument is mandatory.
WIDTH="140"

**HEIGHT**
This value must change based on the interface used. In the case of the standard UI, the value should be set to 40. In the case of the small UI, the value should be set to 24. This argument is mandatory.
HEIGHT="24"

**REGIONNAME**
Specifies a musical region by name. This argument is optional.
REGIONNAME="exposition"

**REGIONNUM**
Specifies a musical region by number. This argument is optional.
REGIONNUM="2"

**BEGIN**
Specifies measure to start playing from. Must be paired with an END argument.
BEGIN="1"

**END**
Specifies measure to stop playing at. Must be paired with a BEGIN argument.
END="5"

**STARTTIME**
Time to start playing - specified as total time elapsed on disc as MinSecFrame. i.e. 4 minutes, 6 seconds and 23 frames is represented as:
STARTTIME="040623"
Must be paired with a STOPTIME argument.
**STOPTIME**

Time to stop playing - specified as total time elapsed on disc as MinSecFrame. Must be paired with a STARTTIME argument.

STOPTIME="041205"

**TYPE**

This may be used to call an instance of the plug-in where a segment file is not necessary, as when passing STARTTIME and STOPTIME arguments. Either SRC or TYPE argument is required.

TYPE="audio/x-audiocd"

**INTERFACE**

Indicates which UI to display on the page. See WIDTH and HEIGHT tags for appropriate settings. This argument is optional and will default to "standard" if absent.

INTERFACE="standard" displays the full size UI.

INTERFACE="small" displays the smaller UI.

---

Standard interface:

Small interface:

Clicking on the CD icon produces a pop-up menu with control commands including, **Play, Stop, Volume +/−, Disc Info, and Help**. Clicking in the display area will also produce a pop-up menu with options specific to playing musical regions directly. This includes the ability to play a section between any two measures.

---

Appendix - User Guide
Note: The plug-in will warn the user if the CD in their drive is not supported by the plug-in’s segment file. An unsupported CD will not play, with one exception. The STARTTIME/STOPTIME arguments will force playback of the specified time regardless of the CD in the drive.
Bibliography


Humes, Malcolm and Rubli, Alex. VCD Sequence Generator http://www.emf.net/~mal/vcdsg/


Newsgroup: comp.sys.mac.programmer.codewarrior (usenet).

Newsgroup: netscape.dev-plugins (private/secure).


References

[1] "Welcome to Netscape"  
(http://www.netscape.com/)

[2] "Microsoft Internet Explorer"  
(http://www.microsoft.com/ie/)

[3] "Status of the W3C httpd"  
(http://www.w3.org/pub/WWW/Daemon/)

[4] "MIDIPLUG"  

[5] "Macromedia: Shockwave Center"  
(http://www.macromedia.com/shockwave/)

[6] "Macromedia - Director Multimedia Studio"  
(http://www.macromedia.com/software/dms/)

[7] "Apple Heat Up QuickTime Plug-In With Hot New Features"  

[8] "MISSION: IMPOSSIBLE"  
(http://www.missionimpossible.com/)

[9] "ID4"  
(http://www.id4.com)
[10] "General MIDI System Level 1 Overview"
(http://www2.midi.org/mma/gminfo.htm)

[11] "RealAudio"
(http://www.realaudio.com/)

[12] "AudioActive"
(http://www.audioactive.com/)

by Elbert McLaughlin


[15] "Voyager CDLink: Listening Room"
(http://www.voyagerco.com/cdlink/)

[16] "VCD Sequence Generator"
(http://www.emf.net/~mal/vcdsg/)
by Malcolm Humes