Digital Content Channels
For the PC

Intel Developer Relations Group

Introduction

The Internet and the World Wide Web have provided a large number of applications that offer real-time delivery of information on demand. The rapid growth of the Internet has been exponential in nature. New Web sites are created each day, trying to capture the attention of surfers on the Internet, and the hours that are spent online are increasing as people discover the vast amounts of information available. It is fast becoming a primary form of entertainment and research for millions.

However, developers face two main obstacles in the development of applications on the World Wide Web. The biggest problem is overcoming the bandwidth limitations that limit the use of rich multimedia in developing applications that are exciting to watch. The second is how to keep information current and relevant to users' interests.

Search engines have become ineffective when returning a large number of hits for a particular topic, and Internet channels have emerged as a superior delivery mechanism for funneling new and exciting content to the end user. Netscape's Netcaster* and Microsoft's Internet Explorer* provide different ways for delivering channels. Just as with cable television, users subscribe to digital channels that contain content relative to their interests. Sports, entertainment, education, and news are just a few of the media channels being created for delivery to the PC via the Internet. Each channel can be personalized, based on the user's preferences. For example, under "sports," the user can select baseball and football to further personalize the delivery of information. Popular Web sites can be subscribed to and then downloaded for off-line viewing. With channels, users narrow their choices and migrate to the most entertaining channels.

Limited bandwidth, meanwhile, translates into limited applications. Web sites are restricted to two-dimensional graphics, text and script-based animations. Rich multimedia, such as full-motion video and 3D graphics that can enhance a Web-based application, require large amounts of bandwidth to be downloaded, and delivered. Delivering these assets on CD- and DVD-ROM can provide much richer content and a better user experience, attracting increased visitors and viewers to an Internet channel. To ensure that the user remains interested in the information the channel is delivering, channels must provide a great user experience.

The driving force in attracting users to a channel is increased revenue. Advertising revenue from the Internet is growing rapidly, and real-time electronic commerce is another source of revenue for companies looking to expand their markets and sales. Channels can reap the economic benefits of both if they can attract viewers.

This white paper describes the key components in creating an exciting digital content channel using key technology developed for the PC. Its purpose is not to recommend a solution, but rather to provide a framework of ideas for developing a compelling Internet channel by integrating high-quality video and audio with interactive and timely information.

We will be looking at an overview of the key components and then follow up with three different Internet channels that apply a variety of different technologies. For developers

unfamiliar with the different technologies presented, the appendix contains reference material. We'll start with a definition of a channel.

What Is a Digital Content Channel?

Simply stated**, a channel is any means of delivering content to the PC.**
A positive benefit of this broad definition is that it does not specify or imply any particular technology or ingredient. Our definition allows the inclusion of a broad number of technologies—old and new. A number of technology vendors—Microsoft, Marimba, Netscape and BackWeb—provide "channel" solutions.
All of their products are good, but they are focused on a narrow band of specific delivery techniques. This paper considers their products and delivery techniques as well as other channel implementation solutions.

Please note another attribute of our definition. The verbs in the definition use the form "ing," which implies a process of action. This is an important distinction for content publishers. The channel concept suggests an ongoing process of content delivery to the user. This means that establishing a channel also means establishing an ongoing *relationship* with the end user. This has not necessarily been the case for publishers who have distributed content solely on CD-ROM, which is packaged and sold in a one-time transaction.

Chapter 1 – Design Requirements, Concepts, Considerations

This chapter describes the design features of channels that you should consider and specify before implementing your project. The target platform, the network infrastructure, the user interface and usage modes are some of the design elements you must consider.

Attributes of Great Content Channels

Our broad definition of a channel does little to define what makes a channel successful. Here are some of the attributes of a channel that will facilitate (though not guarantee) a channel's success.

Compelling Content

First and foremost, the channel must deliver compelling content. Compelling content is **rich**, **timely and relevant**. *Relevant* means delivering the right subject matter to the right users. *Rich* means making the experience deep and rewarding (see Appendix A). *Timely* means delivering content at the right time or delivering something that is new and up to date.

Effective Branding

Successful channels are good at maximizing their brand value or establishing their brand presence. The user interface design can play a big role here. The look and feel of the product should be unique to the brand and consistent within the brand. Elements of the user interface include color selections, navigation paradigms, and usage of screen real estate. Full-screen user interfaces are best, because they make maximum use of the PC resources and exclude the effect of other products and brands.

This presents some problems if the developer chooses to use a Web browser as the user interface. Web browsers can dilute brand presence with the brand of the browser vendor (such as Microsoft or Netscape). Also, the navigation controls in browsers are not unique, which makes it difficult to design the user interface and make effective use of the full screen resolution. This white paper discusses alternative user interface methodologies and provides some of the techniques for providing a full-screen branded interface.

Clear Business Model

The primary question, of course, is how will you make money? A hybrid content delivery requires a unique business model that considers both the traditional CD-ROM business model and Internet business model. The Internet provides revenue streams, such as those from the sale of products, advertising and e-commerce. The revenue for Web sites is dependent on attracting a large number of users. CD-ROMs depend on supplying a product through normal retail channels but typically have a short shelf life. Channels provide a unique opportunity to reach millions of users and extend the life of existing CD-ROM or Web site applications.

User Accessibility and Awareness

Issues often overlooked by developers are *accessibility* and *awareness*. Once you create your channel, how will users find out about it? And then, how easy is it for users to access it?

One of the advantages of the browser method is that it helps to solve the accessibility problem. Almost everyone has a browser; therefore, they have access to the components of your content delivered via the Internet. Additionally, users can request or purchase the CD-ROM content via an Internet transaction, and the CD-ROM can be used to deliver a more compelling user interface, as well as richer content.

The Target Platform and Scalability

According to Moore's Law, the performance of the PC platform for a given price point doubles approximately every 18 months. Ever-faster PCs provide developers with opportunities to introduce new capabilities and experiences that could not be delivered before. These opportunities translate into product differentiation in the marketplace.

The most common mistake that developers make is to underestimate platform capability, particularly over the life of their product. End users tend to invest the majority of their time and money on the PC during the period immediately following the PC purchase. This is the "sweet spot" of end-user interest that developers want to attract, and the best way to attract this interest is to create content aimed at deriving maximum value from these new systems.

Developers should specify their target PC platform to be at least equivalent to the middle- or high-range system at the time their product ships. Then, during the life of the product, the minimum system configuration defined for your product will move down into the lower PC price points. In this way, your product will always have an audience of new PC buyers.

What about the installed base of lower-level but still-capable PCs? Broadening the target market to low-end PCs can be accomplished in many ways. This is where the concept of scalability comes in. *Scalability* is a strategy of providing an alternative experience for the low end. Perhaps this means delivering an alternate set of lower quality audio or video. Or perhaps certain parts of the content are inaccessible to low-end PCs. To effectively implement scalability, it is necessary for the software to detect the PC's performance capabilities. This can be accomplished with the CPUID instruction. (For more information on how to use this instruction, refer to DRG's developer Web site.) The more compelling channels adopt this philosophy: **Aim high; scale down**. Aim your development at the high-end platforms and, if necessary, scale down to meet the capabilities of low-end PCs. This ensures the application delivers the richest possible experience to the end user.

Internet Infrastructure: Bandwidth and Latency

Delivering a rich experience over the Internet can be frustrating to the developer as well as the end user. High-bandwidth connections to the home such as ADSL, satellite and cable modems promise to alleviate this problem. But the fact of the matter is that these technologies are just beginning to emerge, and it will be many years before there is a sufficient market to warrant designing a high-bandwidth delivery strategy. So for today,

and for the next few years, we must face the fact that the majority of users will be connecting to the Internet using POTs rate modems. Highend systems shipped today are typically being configured with 56 Kbps modems. The vast majority of Internet Service Providers can support at least 28.8 Kbps, and they are beginning to make the transition to 56 Kbps. Today, it is safest to assume that the user can connect to the Internet at speeds of at least 28.8 Kbps.

Another quality of service issue that commands much less attention is latency. Defined simply, *latency* is the time for one packet to make a round trip from the end user to the Internet server. Even if the bandwidth required is much smaller than the bandwidth available, there are delays imposed by the Internet that are perceptible to the user.

This is particularly evident with satellite connections. Satellite connections provide very high bandwidth, but the latency is also somewhat high. The packets have to travel a long way—about 44,600 miles round trip. Even at the speed of light, the trip would take about 0.24 seconds.

What does this mean to your design? It is important to consider not only bandwidth delays, but also latency. Every time the user clicks a feature, will the server have to be accessed? If so, the user will wait, even if all the necessary content assets exist locally on the user's system.

Hybrid Delivery

To help solve the bandwidth problem to the home, a new class of PC applications has emerged that combines the timeliness and interactivity of content from the Internet with rich multimedia content delivered via conventional CD-ROMs. These applications are known as *hybrid applications*. The CD-ROM and the Internet are complementary mechanisms for delivering content to end users, as each is suitable for different types of content. This white paper discusses hybrid technologies and components that make it easier to combine the content from CD-ROMs with content from the Internet—to deliver a compelling and seamless experience for the end user.

The hybrid application model uses three basic vehicles for delivering content: live Internet connection, push delivery via the Internet and CD/DVD-ROM. The most bulky and less timely content is distributed on CD-ROM, while the most timely and less bulky content is delivered via a live Internet connection. Push delivery provides a suitable compromise for data that is too bulky for a live connection on the Internet, but cannot wait for the next publishing cycle of a CD-ROM.

Seamless User Interface

Given the multiple delivery strategies of hybrid, the key for application developers is to create an end-user experience that seamlessly combines the content from all three sources. End users should not be burdened with three different user interfaces with which to view content from three different delivery methods. Multiple user interfaces create a partitioning of the content that seems artificial. Next, we'll discuss various user interface technologies and how the developer can create one application to integrate content from all three sources.

Usage Models

With a hybrid content delivery model, it is important to consider which usage models the product supports. Are a CD-ROM and an Internet connection both required for the product to operate? Combining a CD-ROM, push connect and a live Internet connection will provide the richest experience. But will they provide a working (albeit lesser) experience if one is missing? Will it work if there is no push content? Will it work if the user is not connected to the Internet? How will it work without the CD-ROM?

The issues surrounding the supported usage models can also affect the business model for the channel. If advertising is employed, there must be a way to track ad impressions and usage. This is difficult if there is no connection. Is there a commerce element to the business plan? How will transactions be accomplished?

What Makes Content for the PC Rich?

Users will watch channels that are rich in content. Rich content comprises great-sounding audio, full 24- or 30-FPS video, or 3D graphics that make the user's experience realistic. Consumers are used to watching TV, or using software that is very interactive. Today's games provide great 3D interfaces that make them fun to play and immerse users in 3D worlds that are exciting to explore. To keep consumers tuned in to a digital channel, the user's experience needs to be equal to or greater than these other forms of entertainment. The challenge to the developer is the integration of these rich multimedia types with real-time elements that can be delivered over the Internet.

Chapter 2 – Channel Architecture

Channel architecture consists of four major pieces: the client user interface, content management, content delivery and server components. The user interface defines the look and feel of a channel and integrates rich multimedia content into a compelling application. This application is required to deliver a variety of data types that must be displayed within the application. Content management provides the mechanism for managing and delivering scaleable multimedia. Keeping the content new and fresh requires some kind of content delivery. Push technology is the primary means for delivering new content and updates to the application. Content delivery also includes CD/DVD ROM, which provides the large storage necessary to deliver the richest possible content. All of these elements are important in the design and implementation of the channel.

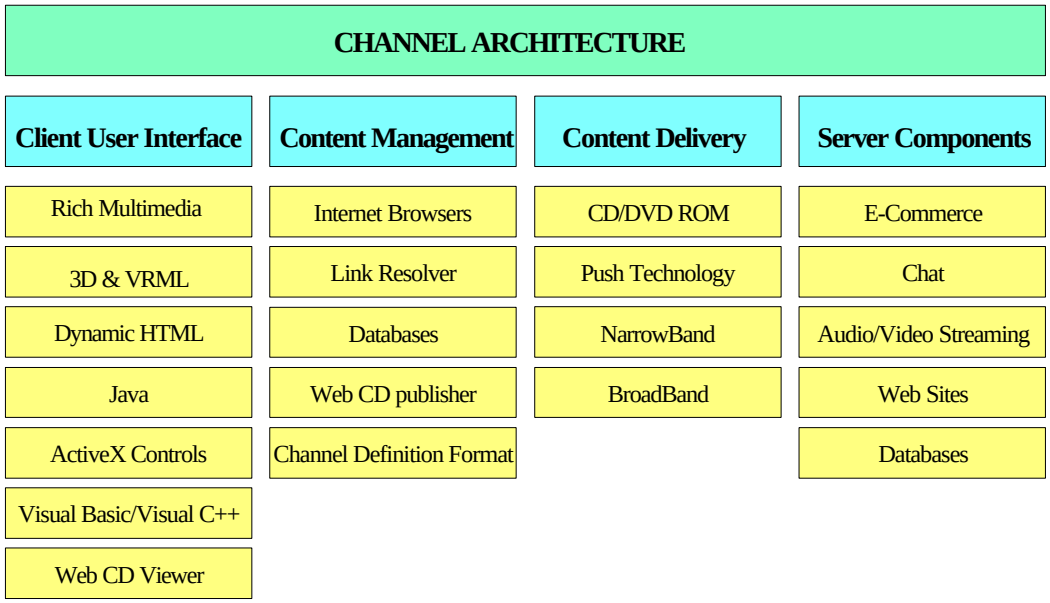

Figure 2-1. Channel Architecture

Client User Interface

The channel's user interface is its most important component, and needs to provide an entertaining experience. Web site user interfaces are typically limited to text and graphics; for channel user interfaces, high-resolution graphics, audio, video and 3D sound can be integrated into an attention-grabbing application.

Unfortunately, performance issues related to delivering content across the Internet have been termed the "World Wide Wait," and adding 3D or interactive elements to the Internet is made difficult by the limited bandwidth available for downloading content. (For information on tools used in designing user interfaces refer to appendix B.) Distributing user interfaces stored on a CD and/or DVD-ROM can enhance the performance as well as

the look and feel of a channel. Of course, the user interface requires that there be some interesting content in the first place.

Content Management

Delivering rich and scaleable multimedia within a channel requires that assets be managed properly. Assets can be delivered based on bandwidth or CPU performance. Video, audio or graphic assets can reside on the Internet, local hard drive or CD-ROM. Content management ensures that the best possible video or audio is delivered to the user regardless of where it is stored. Good content management makes the retrieval of assets transparent, and the interface needs to be designed to dynamically incorporate assets of different quality.

Currently, there are a variety of content management choices. We will explore several methods using Internet browsers, link resolvers, WebCD* publisher and a database. Link resolvers can be used to redirect references on the Internet to assets residing on CD-ROM. WebCD *publisher provides a management tool for delivering Web sites on CD-ROM. Databases can be used as an alternative to manage where content resides, but also deliver content based on user preferences. The method used does not matter; the key is to manage and deliver the richest multimedia to the desktop.

Internet Browsers

Internet browsers manage assets by storing them locally in a cache. Microsoft Internet Explorer 4.0* and Netscape's Netcaster* provide mechanisms for subscribing to channels and uses scheduled polling of the server to pull assets locally onto the user's hard drive. The browser then fetches the Web page, graphic or asset from the cache and displays it immediately. In this way, users can browse Web sites off-line without having to access the Internet. Since the assets are stored locally, performance is greatly improved. However, the primary limitation of this method is that asset downloading is limited by bandwidth constraints, and the richness of the channel is subsequently affected.

Link Resolver

Intel has developed a Microsoft ActiveX* control that arbitrates between assets residing on the CD-ROM, hard drive and the Internet. Assets delivered by push technology and stored on the hard drive can also be resolved using this control. The control contains parameters for specifying the name of the CD-ROM and the push directory location. The CD-ROM name is checked, and if it is not present, the user is requested to plug it in. The control then searches for the assets on the CD or hard drive before going out to the Internet. This flexibility allows the video, audio or graphic asset to be scaleable. Thus, good-quality, 30-FPS video could be stored on CD-ROM, and low-quality, 3–5 FPS video could be stored on the Internet. (Refer to the theater application in Chapter 4 for an example of how to use the Link Resolver control in your application.) A more manageable approach for converting Web sites into CDs is WebCD* Publisher.

WebCD* Publisher

Developers wanting to convert an existing Web site to a hybrid application should consider using a tool like WebCD publisher, which creates a deliverable CD-ROM with

Web pages or assets stored locally, mixed with assets residing on the Internet. This allows developers to manage the process of creating a CD with better-quality video, audio or graphics delivered with the application.

WebCD* publisher provides an easy-to-use tool that is as simple as surfing your site and then redirecting URL links to assets residing on the CD-ROM. The tool also provides a built-in search engine and bookmarking feature for making it easier to navigate through your site. The WebCD architecture includes an ISAPI interface, which makes customization of your CD application easier. A kiosk setting allows for a branded look and feel, with the browser's toolbars and menus eliminated.

The tool provides a comprehensive set of features to create and manage a Hybrid CD application. The game channel application was created using the WebCD* publisher, and the information in Chapter 5 covers step-by-step creation of a WebCD-based application. A potential disadvantage of WebCD* is that the current version cannot be used to integrate Internet and CD-ROM data with data delivered by push technologies. It does, however, have a flexible ISAPI interface which could be used to extend its functionality. Push technology, meanwhile, is useful for delivering content based on user preferences, and databases provide a means to manage assets and target user preferences.

Databases

Databases on the client or the server provide an excellent mechanism for asset resolution. A client-side database can keep track of all assets that could then be queried to determine where the latest version resides. Microsoft Access, Oracle or SQL databases* provide large amounts of information that can be accessed locally or over the Internet. In the music application, Access was used to keep track of audio and video music assets that were stored on the user's CD-ROM or hard disk. The database kept track of where the assets resided—either on CD-ROM or pushed onto the hard drive. The latest database and assets would be delivered on CD and then updated using push technology. The music application using a database to manage content is described in more detail in Chapter 4.

The advantage of using a database is that different types of assets can be mixed as well as retrieved based on user preferences. In this way, assets that are pushed or stored on CD-ROM are only presented to the user if he or she is interested in the content. A heavy-metal rock fan would probably cringe at the prospects of listening to an opera. This dual functionality makes using databases a flexible choice for developers.

Summary

The method chosen for managing multimedia assets depends on the application. Data management of assets ultimately improves the entertainment value of a channel by providing the best possible video, audio or 3D graphics integrated transparently into an easy-to-use interface. Providing better quality and scaleable assets is important for channel developers in order to ensure the success of channels.

Content Delivery

Channels are more compelling when the best possible multimedia assets are made available to the user and there are a variety of choices for the delivery of content to a

channel. Content and software can be distributed on CD/DVD-ROM, and Internet browsers incorporate smart pull to download Web sites.

The emergence of channels in Internet Explorer* and Netscape's Netcaster* allows for the caching of Web site content onto the user's hard drive. Both have mechanisms for subscribing to channels and scheduling downloads of information stored on Web sites. For the delivery of software or assets, push technologies such as BackWeb and Marimba provide flexible solutions. Both are capable of delivering richer multimedia content as well as content based on user preferences.

Internet applications are limited by the relatively low bandwidth of modem connections. Push or pull technology overcomes the limitation of real-time modem connections by downloading content during off hours or in the background. Medium-sized assets or software updates can be pushed, thereby keeping the application updated with the latest software and more compelling content.

CD/DVD-ROM provides a large storage medium for the delivery of rich multimedia content. High quality audio, 3D graphics and video can be delivered using either of these mediums. DVD is the ultimate storage medium, with a minimum of 4.7 gigabytes of storage. This is the equivalent of receiving more than 10 music albums on a single DVD.

In the future, the emergence of broadband will allow a variety of new and exciting applications. Higher bandwidths will facilitate the distribution of software, music and movies to the home to be stored on CD-RW and DVD-RW. With the availability of greater bandwidths, channels will no longer be limited by the size of multimedia or number of assets, enabling higher-quality video and audio to be delivered in real time to the end user.

Narrow Band

Push/Pull from Today's Internet

Push technology is an essential component for delivering compelling and rich channels. Push technology can be used to overcome the limitation of real-time delivery of content over 28.8-Kbps modems. Push technology can also be used to deliver online event announcements as well as advertising, headlines and events. Users can then click on the announcement to invoke a particular channel.

Push technology is available from several companies, including Pointcast, BackWeb and Marimba*, to name just a few. These push servers allow for the delivery of assets or software updates to users after they subscribe to a particular channel. Notification mechanisms are built in that allow users to be notified of incoming updates. In addition, Internet Explorer and Netscape's Netcaster* provide mechanisms for downloading and updating locally cached Web pages. When new information is available, new content is automatically downloaded onto the user's local cache.

From the channel perspective, the updates are "pulled" from the client rather than "pushed" from the server. The client essentially polls the push server for new content and then downloads it if it is available. In our scenario, we are using BackWeb to push InfoPaks that contain urgent notifications for the subscriber.

Notifications alert the user to important information or updates available to the client machine. These notifications consist of small graphic files that serve as advertising

banners. Other types of data that are pushed down include relatively small audio and video assets. Pushing down large audio/visual files that result in long download times is not recommended, and CD-ROM is better suited for delivery of large files.

CD-ROM

The delivery of rich multimedia for a channel on the desktop requires that large assets be stored on CD- or DVD-ROM. Table 4-1 lists the download time required to load the storage capacity of a CD- or DVD-ROM. As you can see, it takes a lot of hours to download the same amount of data from the Internet.

Storage	Capacity	Download Time 28.8-Kbps Modem
CD-ROM	.648 Gigabytes	50 Hours
DVD	4.7 Gigabytes	362 Hours

Table 4-1. Download time of CD/DVD-ROM over 28.8-Kbps modem

CD- and DVD-ROMs can certainly be used to enhance a channel's user interface and content. The emergence of CD-Rs opens up a new distribution mechanism for delivery of software, audio CDs and video content via push technology. CD-Rs could be used to burn downloaded albums or content onto the CD; users could purchase songs on the Internet and then burn them permanently onto a CD. Software could also be distributed and loaded onto the user's hard disk or archived onto CD-ROM.

DVD-ROM

With the availability of DVD-ROMs, higher quality video and audio, such as MPEG2 and AC3, will become the standards for video distribution. Applications distributed on DVD will be able to deliver great-quality video without the storage limits of CD-ROM, and online distribution of movies and music will become a reality as the availability of lower cost read/writable CD- or DVD-ROMs makes possible the downloading of software, movies and music that can be permanently saved.
Eventually, the availability of 1394 Serial Bus or Firewire* will connect the PC with consumer devices such as home theaters and stereos. 1394 Serial bus provides an easy interconnect to pass digital information between these devices. DVD-ROMs or hard disks could easily be plugged in to provide supplemental storage. Hard drives could be added to cache large amounts of information distributed over broadband. Movies, music and software cached on the hard drive could then be permanently stored on a CD- or DVD-ROM. Consumers would then just record them on DVD-Rs or CD-Rs.

Broadband

The availability of low-cost broadband delivery of digital channels to the PC is starting to become available to the public.
ADSL, cable and satellite broadcasts all have advantages and disadvantages relative to digital delivery to the PC. However, the consumer will eventually be able to receive digital data on the desktop at speeds of 1,000 times greater than what is currently available over

a 28.8-Kbps modem. (For a better description of the benefits and limitations of broadband, refer to appendix F.)
With large pipes to the desktop, digital channels will be able to deliver much richer video, data, imaging and 3D graphics to the user in real time.

Satellite Transmission

Satellite data transmission is available today in the form of DirectPC*, which gives users up to 400-Kbps download speed. Live streaming of audio or video can be combined with digital content to deliver real-time content. and rich multimedia assets such as music and video could be transmitted simultaneously to a large number of users. Users can subscribe to channels much as they tune into television broadcasts to receive new content from digital channels.

Standards for satellite and broadband delivery are emerging, and once available, channels can make use of the large bandwidth to provide new digital channels with higher quality audio and video.
So far, we have described content delivered on CD-ROM or hard drives. Another key building block of channels is the delivery of real-time information. Server technology is designed to deliver real-time content to the end user on demand.

Server Components

Building a channel requires the integration of several servers. The number of servers depends on the content being delivered to the user and the business model of the channel. To be able to deliver either stored or real-time content, server solutions need to be scaleable. Channels are made more interesting with the delivery of real-time information from Web sites, streaming of real-time audio/video events and electronic commerce. The Web server is the heart of a channel, and would provide a majority of the real-time information. Electronic commerce servers provide the online stores and software necessary to purchase products from online catalogs. Real-time audio, chat and video servers provide live concerts and interviews for the channel, and push servers provide the delivery of software updates or rich multimedia over the Internet. A conceptual server configuration for a channel which supports all these data types and functionalites is shown in Figure 5-1.

Channel server software consists of the following.

- *HTTP Web Server* – The Web server is typically used to return Web pages or other files using HTTP. Web servers can contain scripts such as Java, CGI, Visual Basic or active server scripts to enhance Web pages.
- *E-commerce Server* – An e-commerce server is required to keep an online store and make online purchases.
- *Database Server* – E-commerce solutions typically involve databases that store customer and transaction information. Depending on the size of the database and number of users on the channel, this can be separate or integrated with the e-commerce solution. Databases store large amounts of related information.

- *Chat Server* – A chat server provides real-time communities where users can meet and talk using voice or audio chat.
- *Real-Time Video/Audio Server* – Real-time events, such as concerts and interviews, can be delivered over the Internet.
- *Push Server* – A push server provides background delivery of software updates or content.
- *Round Robin DNS* – Round Robin DNS servers distribute millions of hits to multiple servers. This allows for scalability and support for a large amount of users.

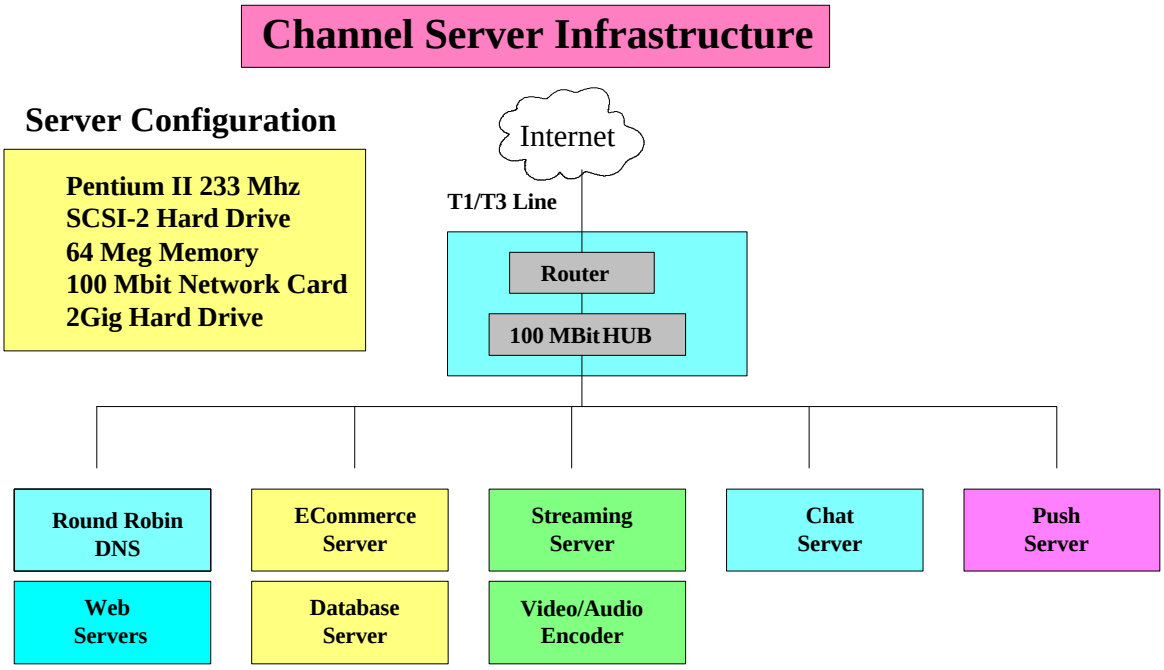

Figure 5-1. Server components

Summary

We have covered the key elements of a channel, including user interfaces, content management, content delivery and server components. The next few chapters look at three applications: a music, a theater and a game channel. Each channel takes a different approach in the design of the major components, and the applications demonstrate the integration of technologies and concepts that go into constructing a successful channel.

Chapter 3 - Music Channel

The Music Content channel application gives the user a jukebox interface for previewing CD-albums and songs. This provides a great user experience by storing high-quality audio and video on CD. The application delivers high-quality music WAV files for playback off the CD, and music videos captured in Indeo Video 5.0 format are provided as well. The channel plays back music to the user based on his or her preferences for rock, jazz, new age, classical and other popular formats.
The music application, graphics and current audio and video samples are delivered on CD-ROM. The front end includes both a music and advertising database to target music based on the user's preferences, while the database, music videos and audio tracks are kept current by a BackWeb server. The database can reside locally or be stored on a server on the Internet. In this case, the database is stored locally on the user's hard drive.
Databases are one of the methods for providing content management of a channel. Visual Basic 5* was selected as the programming language to integrate all the key elements of the application. Since the application is delivered on CD, the client interface can be rich with great graphics, audio and video integrated into a high tech interface.

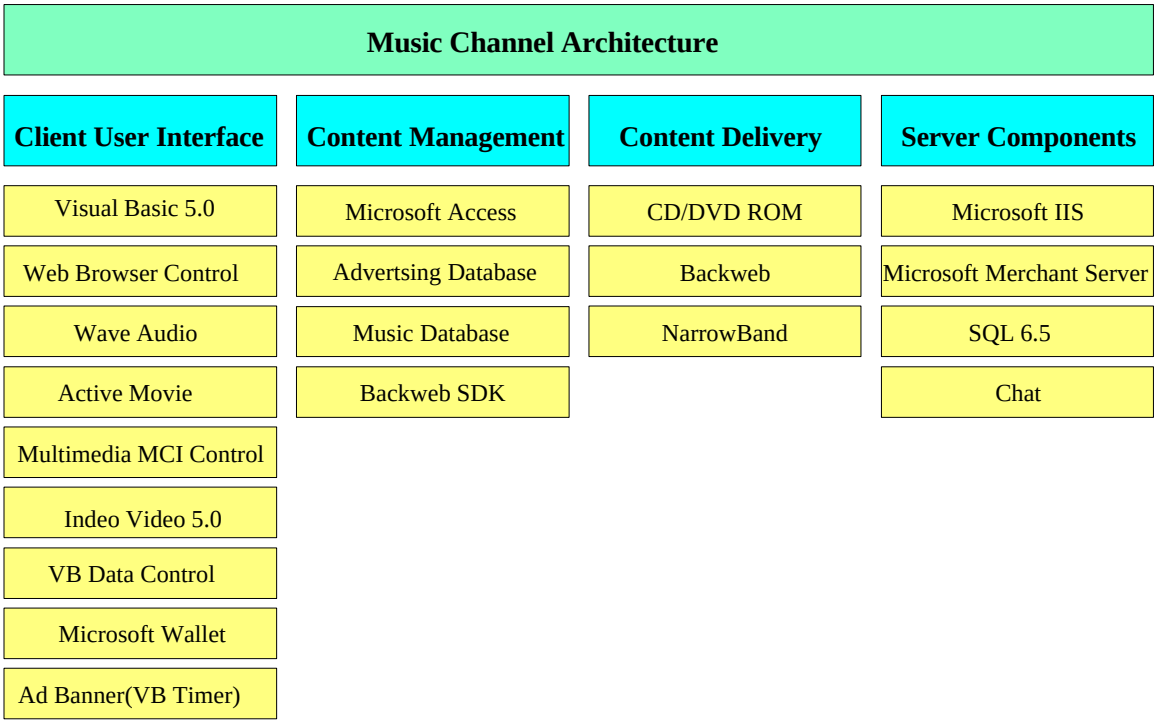

Figure 3-1. Music Channel Architecture

Client User Interface

Figure 3-2 Music Channel User Interface

The application is written in Visual Basic 5.0* using standard Visual Basic. A variety of controls are used to develop the application: Data, Label , Multimedia MCI , WebBrowser , and PictureBox. The interface includes a high tech graphic design stored as a GIF file, which provides the background for the application. Visual basic applications are created by inserting controls into a form and then writing scripts for user events. The Label and Picture Box controls provide the dynamic elements of the application. PictureBox controls embed the graphic elements such as the background, buttons, and album covers arranged on top of the background. Label controls contain the text information about the artist, or CD.

The client user interface links information from the database to Label and PictureBox controls to display information about artists, CD albums and tracks. DataControls are inserted into the form to create a link to the actual database where information is stored. User interface elements are then tied to the current record in the database by specifying

the DataControl and Datafield labels in the PictureBox or Label control's properties. An advertising database is searched to provide banners which can be targeted to a user's interests. An ad-banner interface with rotating ads was implemented using a PictureBox control and Timer.
Four buttons on the user interface navigate through the music CD albums and the tracks within each album. By selecting the buttons the application pulls up the next album stored in the database and then updates all the user elements. The next step is to load the appropriate audio track or song to play when a user presses one of the navigation keys.

Loading a Song

The use of a database adds the flexibility to support a large variety of audio sources, determine what audio source is being played and where the source resides. Controls are then made visible or invisible depending on whether audio or video is being played back. Assets are stored by *RecordingID* and *TrackNumber*. The format of the file, whether WAV or AVI, is also stored in the database to determine how the song is played. Other formats could easily be added, such as MIDI, AC3 and other audio compression files.

Loading a Song

```
MMControl1.Command = "close"
Image7.Visible = True
ActiveMovie1.Visible = False
If (Albums.Recordset("Format") = "HD") Then
MMControl1.DeviceType = "WaveAudio"
MMControl1.filename = "c:\demo\music\assets\" & Albums.Recordset("RecordingID") &
"-" & Tracks.Recordset("TrackNumber") & ".wav"
MMControl1.Command = "open"
MMControl1.Command = "Play"
End If
If (Albums.Recordset("Format") = "CD") Then
MMControl1.DeviceType = "WaveAudio"
MMControl1.filename = lpRootPathName & "demo\music\assets\" &
Albums.Recordset("RecordingID") & "-" & Tracks.Recordset("TrackNumber") & ".wav"
MMControl1.Command = "open"
MMControl1.Command = "Play"
End If
If (Albums.Recordset("Format") = "CDV") Then
WebBrowser1.Visible = False
Image7.Visible = False
ActiveMovie1.Visible = True
ActiveMovie1.filename = lpRootPathName & “demo\music\assets\”
Albums.Recordset("RecordingID") & "-" & Tracks.Recordset("TrackNumber") & ".avi"
End If
Image7.Picture = LoadPicture("c:\demo\music\assets\" &
Albums.Recordset("RecordingID") & "-" & 1 & ".gif")
WebBrowser1.Visible = False
```

Figure 3-3 Code for loading songs

Data Controls provide a link to a database. The data set control for the music database is initialized by setting the control’s properties Name to *Music.mdb* and the record name to “records.” Before loading a song, the *Format* database field is tested and determines the type of file to play. The *Format* field is set to HD, CD and CDV, and corresponds to Hard Disk audio, CD audio and CD video, respectively.
The code for playing an audio or video track is shown above. Wave audio is played using the Multimedia MCI control. The control works by specifying the Device type as WAVAudio and then specifying the filename for the audio file. Two MCI commands are then issued to open the file and then play it. Video files residing on CD are played using ActiveMovie. The ActiveMovie control is first made visible and filename set to the

appropriate AVI file. When playing back a video file, the album cover is made invisible. The album cover is loaded regardless of the type of asset but is only made visible by audio clips. The same technique of making elements visible or invisible is used to provide Web access using the WebBrowser control.

WebBrowser Control

The Standard WebBrowser control allows for the integration of live Web pages into a Visual Basic application. In the music application, the control is used to buy albums through an e-Commerce server when the user has made a purchasing decision. The WebBrowser control can embed Web pages on the Internet into the Visual Basic's user interface. In the music application, the WebBrowser control is made visible when the purchase button is pressed. This pulls up the music purchase Web site page containing the Microsoft Wallet* and address controls. Loading another song by hitting one of the scroll buttons makes the WebBrowser control invisible by setting the WebBrowser1.Visible attribute to "FALSE." Using advertising banners, the same method could be used to link to live audio concerts broadcast on the Web, electronic commerce servers, Web sites or advertisers. The banners could announce concerts, live events or CD-album sales.

```
WebBrowser1.Visible = True
 WebBrowser1.Navigate "134.134.246.52/musicpay.htm"
```

Figure 3-4 WebBrowser Code for accessing the Internet

Advertising Banners

Advertising banners use a simple Visual Basic timer that can be set to display new graphics periodically. At given intervals, it simply moves through the advertising database and displays the next record. When the image is clicked, the URL for the advertising banner is invoked and the Web site can be displayed according to the information being shown. Although not implemented, an SQL search could be done based on the user preferences. The next section goes into more detail about databases.

```
Dim Count As Integer
Count = Count + 1
Advertising.Recordset.MoveNext
If (Advertising.Recordset.EOF = True) Then
Advertising.Recordset.MoveFirst
Else
End If
If (Count < MaxRecords) Then
Count = 1
End If
AdvertisingPic.Picture = LoadPicture("c:\demo\music\assets\" &
Data4.Recordset("adname") & ".gif")
```

Figure 3-5 Advertising Banner Code using the Timer Control

Content Management

Content management in the music application was implemented using a Microsoft Access Database*. The previous section already covered asset management using the database. This section describes scrolling through albums and updating of the database information on the user interface.

Microsoft Access Database

The Microsoft Access* database provides an easy-to-use framework for developing database applications. Integration with Visual Basic 5.0* makes it easy to develop simple client-side databases for extracting information. In this case, four databases were defined: artist, album, track and advertising. These databases are linked to provide the user with a potentially large amount of music information that matches his or her interests.

Relational Databases

A relational database consists of data stored in tables by columns and rows. Columns are essentially *fields*, and rows are referred to as *records*. Fields define the data that resides in a particular database (text versus dates for example). Each row in a database contains information about a particular field, and it is expected that each row be unique. Databases are normally queried using SQL statements, which search a particular database and return the records or rows that match the search criteria.

Local Database

The music application provides a local database of music albums, tracks and artists to present information to the user corresponding to the selected track. The user can browse through the local albums or tracks in the current database, and the application will display an album cover when playing audio tracks, and video clips for a music video. Music audio tracks can also be pushed from a BackWeb server to the client. This occurs in the background with no intervention from the user. The user can subscribe to channels based on preferences such as rock and country, providing both targeted advertising and delivery of music that the user is most interested in. With BackWeb, the local database is updated with a new one, and a new audio track is delivered over the Web and played on the user's local hard drive. The user receives the best quality audio from either the CD or the HD.

Scrolling Through Albums

The music application allows the user to scroll through albums. The Music Collections database uses the standard template provide by Microsoft Access*, and consists of three tables: a record, tracks and an artist table. These store information about a particular album, name of artist and record label, and the information is displayed and navigated using the standard data control provided with Visual Basic 5.0. Setting up a data control consists of setting up the *Connect* property to set the database (Access in our case), the *Database name* (set to *music.mdb*) and finally the record source. Once these are set, it is relatively easy to traverse the records in the database to extract needed information.

A record set can represent the information stored in a base table or the results of an SQL search. In our application, the Album table is navigated by using the *MoveNext* and *MoveFirst* functions. Before moving to the last record, the EOF flag is checked in order to reset the database to the first entry when the user is at the last record.

```
Albums.Recordset.MoveNext
If (Albums.Recordset.EOF = True) Then
Albums.Recordset.MoveFirst
Else
End If
Tracks.RecordSource = "Select * From Tracks Where RecordingID=" &
Albums.Recordset("RecordingID")
Tracks.Refresh
Artists.RecordSource = "Select * From [Recording Artists] Where RecordingArtistID="
& Albums.Recordset("RecordingArtistID")
Artists.Refresh
Load_Song
```

Figure 3-6 Album Scrolling Albums Code

After moving to the next record in the albums, the Tracks and Artist databases are updated to be set to the appropriate track and recording artist. First an SQL search is made of the records, with the Albums set to the recording ID. The record is then refreshed in the application to update the information displayed in each of the data displayed in the application. Visual Basic allows for resources to be set to a particular field in a database. and as the database records are moved to the next field, the information displayed in label controls is updated as well. The album name, artist name, recording studio, name of track and price information is all updated as each database is first searched and then refreshed. Databases provide a lot of flexibility to add supplemental information about the artist, CD-album, or songs. These records can be searched based on the user's preferences for artists or music genre.

Content Delivery

The WAV files, music.mdb and advert.mdb databases are delivered on CD-ROM with rich multimedia assets. New WAV files and both databases are updated using push technology and ensures users are always updated with the latest songs and CD-albums. Electronic distribution of music is one of the applications made possible with push technology.

CD Music Videos and Albums

The CD-ROM for the music channel contains the client software and multimedia assets for the music channel. The CD delivers music videos using Indeo Video 5.0 and high quality audio tracks using 44Khz 16-bit wave audio. High-resolution (800x600), 24-bit graphics enhance the user interface to provide a high resolution graphic interface. The multimedia assets provide a much better user experience than possible over the Internet, and a BackWeb* server is used to provide updated content.

Pushing New Content

The BackWeb client needs to be connected to the BackWeb server to receive new audio tracks. The BackWeb server delivery mechanism works by sending what is called an InfoPak (Appendix E describes in detail the BackWeb server solution.) We have implemented software that transfers the files in this InfoPak into the appropriate directory for the music channel. (Refer to Appendix F for a description of the push controller designed using the BackWeb SDK*.) The files in the InfoPak contain a new database of audio tracks and albums as well as the audio content. The advertising database and ad-banners can also be updated using the same mechanism, and the files delivered by BackWeb could be dependent on whatever music channels the user subscribes to: rock, country and so on.

Indeo Audio was used to compress the audio files for delivery using Push. At 8:1 compression, audio files can be delivered with minimal reduction in quality. A three-minute song with good quality, for instance, could be delivered in about 20 minutes. With compression, push technology provides a viable solution for electronic distribution of music.

Electronic Music Distribution

Music samples and content could be delivered in a variety of compression formats, depending on application needs. Compressing the music can deliver a lot of music samples, assisting the user in album buying decisions. Complete, uncompressed versions of songs could also be downloaded, although it would take close to an hour. (On the other hand, , music fans would likely deem the wait worthwhile, in order to obtain new songs from their favorite artists.)

Album art and band pictures could also be delivered. Supplemental album art could be distributed as a benefit to subscribing to a channel, and band or concert pictures could be delivered to fans, who may want to print and store them in digital scrapbooks. The push server would deliver software updates to the channel's user interface.

Security is a major concern with electronic distribution, and standards for encryption and digital watermarking of music content -- such as the International Standard Recording Code(ISRC) for watermarking or RSA for encryption -- are evolving. Artists receive a continued revenue stream from royalty payments from broadcast or distribution of their songs through licensing organizations, and online distribution of music promises to increase the revenues for both artists and music distributors. However, music channels must address encryption, digital watermarking and royalty collections before distribution becomes a reality.

Server Components

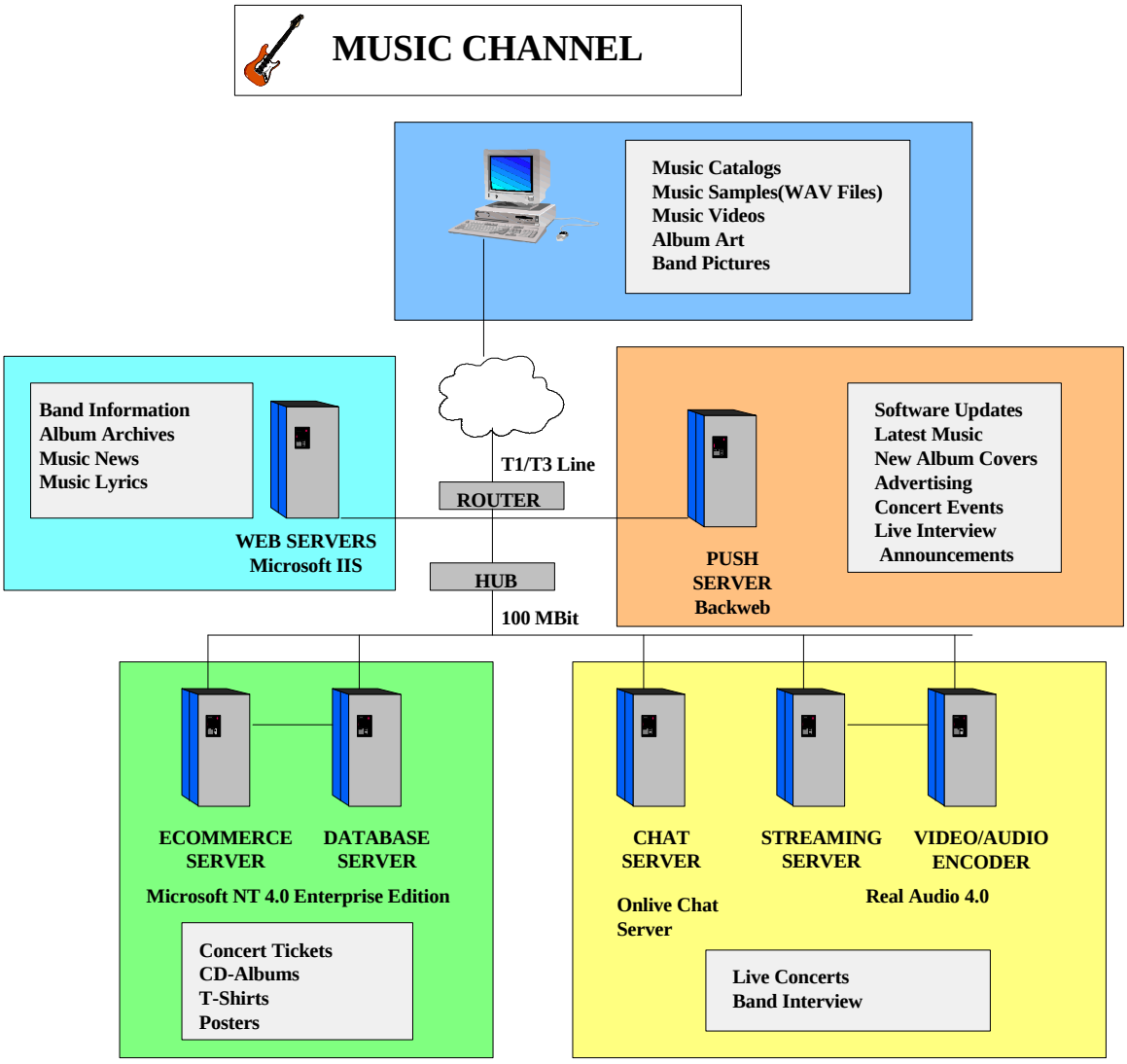

Figure 3-7 Music Channel

Web Site

A theoretical music channel with all the server elements is shown in Figure 3-7. The Web site contains updated or additional information such as: the latest music news, concert information, album releases, lyric, sheet music and band information.

Live Concerts and Band Interviews

To promote merchandise and album sales, live events such as concerts and band interviews can be broadcast across the Internet. A chat server could be used as a way for fans to ask questions about their favorite bands. Real-time data like audio or video and chat provide a sense of community in the music world. Advertising targeted at concert demographics could sell concert albums or T-shirts at special prices during events, thus providing additional revenue streams for existing albums and merchandise.

Electronic Commerce

The e-commerce server enables users to purchase the latest albums, concert tickets, and other merchandise. Using Microsoft Wallet or CyberCash*, consumers can easily purchase merchandise through secure transactions. The music application also provides for electronic commerce using Microsoft's Commerce Server*. This was used as an example because the server handles order processing, order tracking, inventory, credit card transactions, product information, pricing, shipping and so on required by electronic commerce. The user then has the ability to listen to sample audio tracks distributed either on CD or updated via the Web, and then purchase the CD-albums online.

The database server stores information such as: online product catalogs, customer purchases, music CDs by category, advertising content based on demographics, and user selections and preferences.

Embedding Microsoft Wallet* Into Web Pages

Microsoft Wallet* is intended to be filled out by the user so that his or her address and credit card information are already recorded prior to making a purchase. Multiple addresses and credit cards can be entered in each control. The credit card information is password-protected at the client to prevent anyone else from making a purchase. Electronic commerce within channels requires integration of the product information into a simple Web page. Once the user presses the Buy button, the order can be processed by the Microsoft Merchant Server* or with a similar e-commerce server solution.

```
<object id="addrSelector"
classid="clsid:87D3CB63-BA2E-11cf-B9D6-00A0C9083362"
codebase="MSWallet.cab#Version=2,1,0,1374"
align="left" border="0" hspace="20"
width="154" height="123">
</object>
<object id="paySelector"
classid="clsid:87D3CB66-BA2E-11cf-B9D6-00A0C9083362"
codebase="MSWallet.cab#Version=2,1,0,1374" align="baseline"
border="0" hspace="20" width="154" height="123">
<param name="AcceptedTypes" value="*:*">
<param name="Total" value="134.50">
</object>
```

Figure 3-8 Code to Embed Microsoft Wallet* into Web Pages

Summary

The music channel delivers a software application on CD-ROM to sell CD-albums and promote music artists. Music video and songs are delivered using high-quality Indeo Video 5.0 and Wave audio music samples, and Indeo Audio is used to provide compressed audio for faster delivery using push. Visual Basic 5.0 provides a variety of controls to develop applications rapidly and deliver the best possible user experience, while WebBrowser control provides the integration of the client application with server applications such as chat, electronic commerce and live audio streaming. Databases and push technology allow the music being delivered to be targeted to the users interests. These are the key elements that deliver an exciting music channel that will attract fans and increase CD-album sales.

Chapter 4 - Theater Channel Application

The theater application provides a preview of new movies being released in theaters. The application uses dynamic HTML to deliver high-quality video clips and information about released movies, and the application makes use of several key technologies: Indeo® Video 5.0 is used to deliver high-quality 30-FPS video on CD-ROM, Content management of the channel is done using Active Channel and a Link Resolver, and Microsoft's Channel Definition Format is used to provide off-line browsing and update Web pages*. Web sites, meanwhile, deliver the latest movie information and trivia, videos are sold through e-commerce solutions on the Web, and push servers are used to push new content as well as advertising.

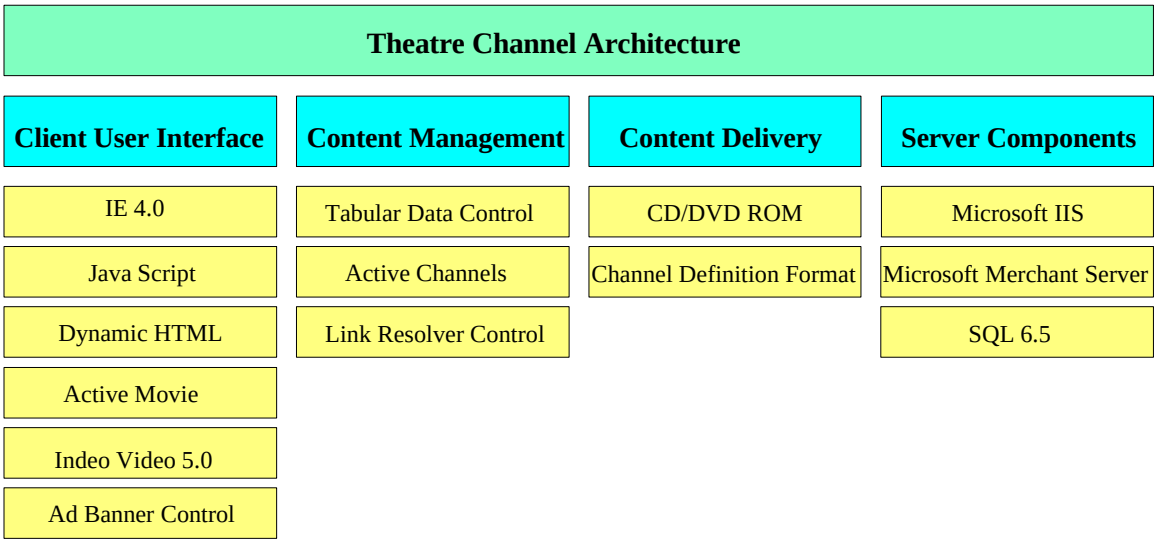

Figure 4-1. Theatre Channel Architecture

Client User Interface

The client interface for the theater application was written entirely in dynamic HTML and Java Script*. ActiveMovie is used to play Indeo Video 5 AVI movie trailers within the HTML page. A Java-based ad-banner control provides rotating ads to link to live Web sites The application takes advantage of the object oriented nature of the newest browsers and Java scripts to provide a robust application.

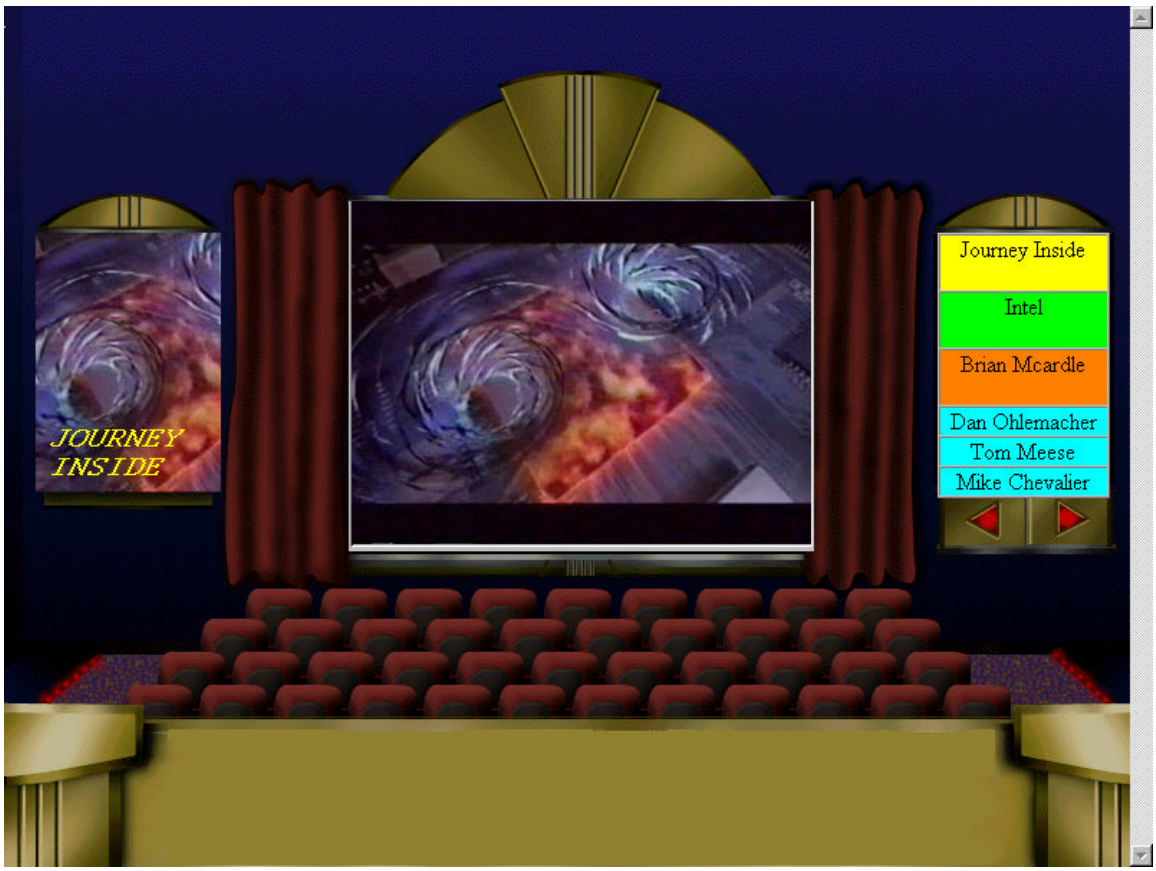

Figure 4-2 Theatre Channel User Interface

Dynamic HTML

Several features of Dynamic HTML were used to provide a better graphical interface. Dynamic HTML extends standard HTML by adding an object model. Objects in a Web page can be labeled and scripts written to dynamically change the objects. Special effects can be applied to graphics or text to liven up a static Web page. Dynamic HTML allows for dynamic positioning of objects within the browser window. This feature was used to properly position graphic buttons, active movie control and ad banners on top of the background. Controls and graphic elements where labeled to allow scripts to operate on these objects based on the interaction.

Active Movie

The active movie control is used to playback the Indeo video 5.0 video clips. The function DoPlay for changing the URL reference within the movie control is shown in Figure 4-2. This function is called whenever a navigation button is selected. The ShowControls and ShowDisplay are to turn off the multimedia controls and display

counter within the active movie control. Autostart is set to false to run the movie clip as soon as it is loaded. The filename is set by calling DoPlay which calls the Link Resolver for the appropriate URL reference whenever the navigation buttons are hit.

```
<OBJECT ID="ActiveMovie1" WIDTH=320 HEIGHT=240
CLASSID="CLSID:05589FA1-C356-11CE-BF01-00AA0055595A" border="1"
style="position: absolute; top: 136; left: 240">
      <PARAM NAME="_ExtentX" VALUE="106">
      <PARAM NAME="_ExtentY" VALUE="106">
      <PARAM NAME="ShowDisplay" VALUE="0">
      <PARAM NAME="ShowControls" VALUE="0">
      <PARAM NAME="AutoStart" VALUE="-1">
   </OBJECT>

function DoPlay(sURL){
var sNewLink
sNewLink = LnkRslv1.Resolve(sURL)
ActiveMovie1.FileName = sNewLink
}
```

Figure 4-3 Active Movie Control

Full-Screen Mode

The theater channel is run in full-screen mode to provide a fully branded interface without the normal browser controls or menus. Setting the Internet Explorer browser to full-screen mode requires the use of a starting HTML page that launches the theater page in full screen. The simple script to create a full-screen branded interface is shown below:

```
<script language=vbscript>
window.open "http:\\134.134.246.52\theater\theater.htm", "",
"toolbar=no, status=no, location=no,directories=no, fullscreen=yes"
window.close
</script>
```

Figure 4-4 Full Screen Mode Script

The toolbars, status bar, location window and directories are set to “no” so they will not be displayed. The full-screen parameter is set to “yes” to set Internet Explorer to full screen.

Advertising Banners

The theatre application includes advertising banners for rotating the latest movie releases. Microsoft FrontPage* supports a variety of Java applets to support advertising banners. Ads can include effects such as blinds, dissolves, box in or box out, and the effect is specified in the rotator effect parameter. The time specifies the number of seconds that the ad should be displayed. Finally, the URL parameter specifies the link to jump to when the ad-banner is pressed. The advertising banners can be of any width or height, and the banner is embedded on top of the background graphic by using dynamic positioning. The position of the banner is specified as absolute and offset 160 pixels from the top and 26 from the left.

```
<applet code="fprotate.class" codebase="_fpclass/" width="120" height="175"
style="position: absolute; top: 160; left: 26;">

  <param name="rotatoreffect" value="boxOut">
  <param name="time" value="5">
  <param name="url" value="http://www.sony.com/" valuetype="ref">
  <param name="image1" value="mibb.jpg" valuetype="ref">
  <param name="image2" value="afob.jpg" valuetype="ref">

 </applet>
```

Figure 4-5 Advertising Banner

Content Management

Content management in the movie channel uses Microsoft's Tabular Data Control and Intel's Link Resolver. The TDC acts as a database for storing movie information such as the actor names, director, price and movie genre. The Link Resolver Control resolves whether the asset resides on the hard drive, CD or on the Internet. To update the theatre application, a sample Channel Definition File is created to periodically search the Web site for the latest content. The application is centered around the Tabular Data Control and moves between records stored in the database text file.

Microsoft's Tabular Data Control

The Tabular Data Control retrieves data within a delimited text file and displays it in a table in the HTML page. Each data element in the file is defined as the first line and separated by commas. Each additional line specifies the data set by a special delimiter. Each data element can then be assigned to a table entry within a dynamic HTML page or even an object.

The theater application stores information about each movie in the text file. The database file used in the theater application is shown below:

```
record|movie|studio|reldate|director|actor1|actor2|actor3|avifile|price|Type
!1!|!Air Force One!|!Columbia Pictures!|!7/2/97!|!Wolfgang Peterson!|!Glenn
Close!|!Harrison Ford!|!Gary Oldman!|!e:\airforce one.avi!|!39.95!|!Action!
!2!|!Men In Black!|!Columbia Pictures!|!7/4/97!|!Barry Sonnenfield!|!Will Smith!|!Tommy
Lee Jones!|!Linda Fiorentino!|!e:\mib.avi!|!39.95!|!Sci-Fi!
!3!|!Starship Troopers!|!Tri Star!|!7/4/97!|!Paul Verhoeven!|!Casper Van Dien!|!Michael
Ironside!|!Rue Mcclanahan!|!e:\starship Troopers.avi!|!39.95!|!Sci-Fi!
```

Figure 4-5 Movie database stored in TDC Text Format

The definition of a movie control consists of specifying the data file, the text qualifier and the field delimiter. The initial record filter is set to 1, the first record in the database.

Moving Between Records

The theater application stores information about each movie in the text file. Dynamic HTML is used to process mouse clicks on the object. To move through the records, each of the button objects—*browseleft* and *browseright*—specify one click to call the functions *previousrecord* and *nextrecord*. The code used to define the TDC control (labeled Movie) and move between records is shown below.

On pressing the *browseleft* and *browseright* buttons, the record number is checked and incremented/decremented or set to the first/last record. The text database file is then searched for the appropriate record number with the *Movie.Filter* call. The data is updated with the *Movie.Reset* call.

This simple code allows for the inclusion of simple database records to dynamically change the text displayed in tables. Any object can be defined and tied to a record defined in the Movie TDC. This code is shown in Figure 4-7.

```
<object id="Movie" width="0" height="0" style="position: absolute; top: 0; left: 0"
classid="CLSID:333C7BC4-460F-11D0-BC04-0080C7055A83">
 <param name="TextQualifier" value="!">
 <param name="FieldDelim" value="|">
 <param name="DataURL" value="movie.txt">
 <param name="UseHeader" value="True">
 <param name="Filter" value="record=1">
</object>

browseleft.onclick = nextrecord
browseright.onclick= previousrecord

function previousrecord() {
if(recordno > 1)
{
recordno--
}
else
{
recordno=3
}

Movie.Filter="record=" + recordno
Movie.Reset()
DoPlay("http://134.134.246.52/" + recordno + ".avi")
}

function nextrecord() {
if(recordno < 3)
{
recordno++
}
else
{
recordno=1
}
Movie.Filter="record=" + recordno
Movie.Reset()
DoPlay("http://134.134.246.52/" + recordno + ".avi")
}
```

Figure 4-6 Tabular Data Control and Script for moving between records

```
<table id="shop_tbl" border="1" datasrc="#Movie" cellspacing="0" cellpadding="0"
width="120" style="position: absolute; top: 158; left: 650;" height="145">
  <tr>
    <td bgcolor="#FFFF00" height="12"><p align="center"><big></big><span
datafld="movie"></span>
    </td>
  </tr>
  <tr>
    <td bgcolor="#00FF00" height="12"><p align="center"><font face="Georgia"
color="#0000FF"></font><span datafld="studio"></span></td>
  </tr>
  <tr>
    <td bgcolor="#FF8000" height="12"><p align="center"><font face="Georgia"
color="#000080"></font><span datafld="director"></span></td>
  </tr>
  <tr>
    <td bgcolor="#00FFFF" align="center"><font face="Georgia"
color="#000080"></font><span datafld="actor1"></span></td>
  </tr>
  <tr>
    <td bgcolor="#00FFFF" align="center"><font size="24" face="Georgia"
color="#000080"></font><span datafld="actor2"></span></td>
  </tr>
  <tr>
    <td bgcolor="#00FFFF" align="center"><font face="Georgia"
color="#000080"></font><span datafld="actor3"></span></td>
  </tr>
</table>
```

Figure 4-7 Table definition linking data fields to TDC database

Link Resolver ActiveX Control

The Link Resolver is a simple ActiveX control that redirects a URL reference from an Internet access to a local CD or hard drive. The control has several parameters. The first is the CD-ROM name, which is verified to ensure that the CD containing the asset is the correct one and is inserted into the CD-ROM drive. In addition to specifying the parameters, the link control is called with the appropriate URL to be checked, and then it will resolve the appropriate reference. The function for the Link Resolver is called when processing database records. The ActiveMovie.Filename is set to the URL consisting of the TCP address for the server, the record number and the ".AVI" extension. The appropriate movie assets are stored on the CD and played accordingly.

```
<OBJECT ID="LnkRslv1" WIDTH=0 HEIGHT=0
 CLASSID="CLSID:4E401503-7DCB-11D0-A060-00AA00C0F5D6">
 CODEBASE="lnkrslv.ocx#Version=1,0,0,001">
 <PARAM NAME="_Version" VALUE="65536">
 <PARAM NAME="_ExtentX" VALUE="0">
 <PARAM NAME="_ExtentY" VALUE="0">
 <PARAM NAME="_StockProps" VALUE="5">
 <PARAM NAME="CD-ROMLabel" VALUE="Hybrid">
 <PARAM NAME="LinkName" VALUE="">
        <PARAM NAME="PushDataDir" VALUE="c:\demo\movie\assets">
 <PARAM NAME="Text" VALUE="">
 </OBJECT>
```

Figure 4-7 Link Resolver Control and Code to update Active Movie Filename

CDF Channel Definition Format

The theater application can be turned into an Internet Explorer* channel by adding a channel definition format (CDF) file for caching the theater HTML pages on the user's hard disk. This improves performance and allows for off-line browsing of content. Defining a channel is a simple process. The steps involved are:

Set up the channel, specifying channel bitmaps, channel title and the abstract describing the channel.

Specify the pages to be included in the channel.

Specify whether the channel pages should be cached or not.

Define the channel as an e-mail notification, screen saver, desktop component, channel or hide it.

Schedule the channel for updates.

There are several tools for defining the CDF format. Frontpage97* from Microsoft provides a simple wizard for defining a channel. The final output for the wizard is shown below. The channel is described as the Theatre channel and includes the two Web pages in the theater application.

```
<?XML Version="1.0" Encoding="iso-8859-1" ?>
<Channel HREF="start.html" BASE="http://134.134.246.52/" SELF="channel.cdf">
        <A HREF="http://134.134.246.52/start.html">
        </A>
        <Title>Theatre Channel</Title>
        <Abstract>The Theatre channel provides the latest previews of the latest
movies</Abstract>
        <Logo HREF="movie.bmp" Style="Image" />
        <Logo HREF="moviei.bmp" Style="Icon" />
        <Schedule>
                <IntervalTime DAY="7" />
        </Schedule>
        <Item HREF="start.html" Precache="Yes">
                <A HREF="http://hybridtest2/start.html">
                </A>
                <Title>start.html</Title>
                <Usage Value="Channel">
                </Usage>
        </Item>
        <Item HREF="theater.htm" Precache="Yes">
                <A HREF="http://hybridtest2/theater.htm">
                </A>
                <Title>This is sample text</Title>
                <Abstract>The Theatre channel provides the latest releases for
preview.</Abstract>
                <Usage Value="Channel">
                </Usage>
        </Item>
</Channel>
```

Figure 4-8 Sample Channel Definition Format

Content Delivery

The theatre channel distributes movie trailers on CD-ROM. The application uses the Channel Definition Format and Active Channels to keep the Web pages updated. Push Technology could deliver the latest movies based on the user's preferences.

CD- or DVD-ROM

The CD-ROM delivers 30-FPS videos of movie trailers and the interface for the movie application. With either Indeo Video 5.0 or MPEG2 video, the movie trailers would

be high-quality. Users could get quarterly updates of the latest movies being released on the CD-ROM.

DVD would be the ideal storage choice for delivering movies encoded in MPEG2 video and AC3 audio. With 4.7–17 GB of storage, 144 minutes of video can be stored on a DVD. That represents at least one full-length movie, but for the movie application, that would represent 144 trailers at 1 minute each in length. If it were MPEG1 or Indeo Video 5.0, more than a 1,000 trailers could be stored on a DVD. Full-length movies could be distributed with software included to run the channel as well. The application could be delivered with every DVD movie purchased, enhancing the value of the DVD over just purchasing the movie, as consumers get the benefits of a DVD movie, plus the added features and information a channel application could provide.

Push

The push servers could update users subscribed to the channel with the movie trailers of recent releases. Indeo Video 5.0 could be used to provide scaleable video downloads online. MPEG2 clips take considerable bandwidth, but over broadband they could easily be distributed. Local demographics could be used to deliver theater locations where the movie is being shown. The movie trailers themselves represent the best form of advertising, and ad banners could be pushed showing specials for online VHS or DVD movie purchase.

Push servers could also deliver movie trailers based on the user preferences, such as movies in which favorite actors appear. The user is then receiving the movies that interest him or her the most. Advertising could be tailored to highlight the latest releases and specials on movie merchandise.

Server Components

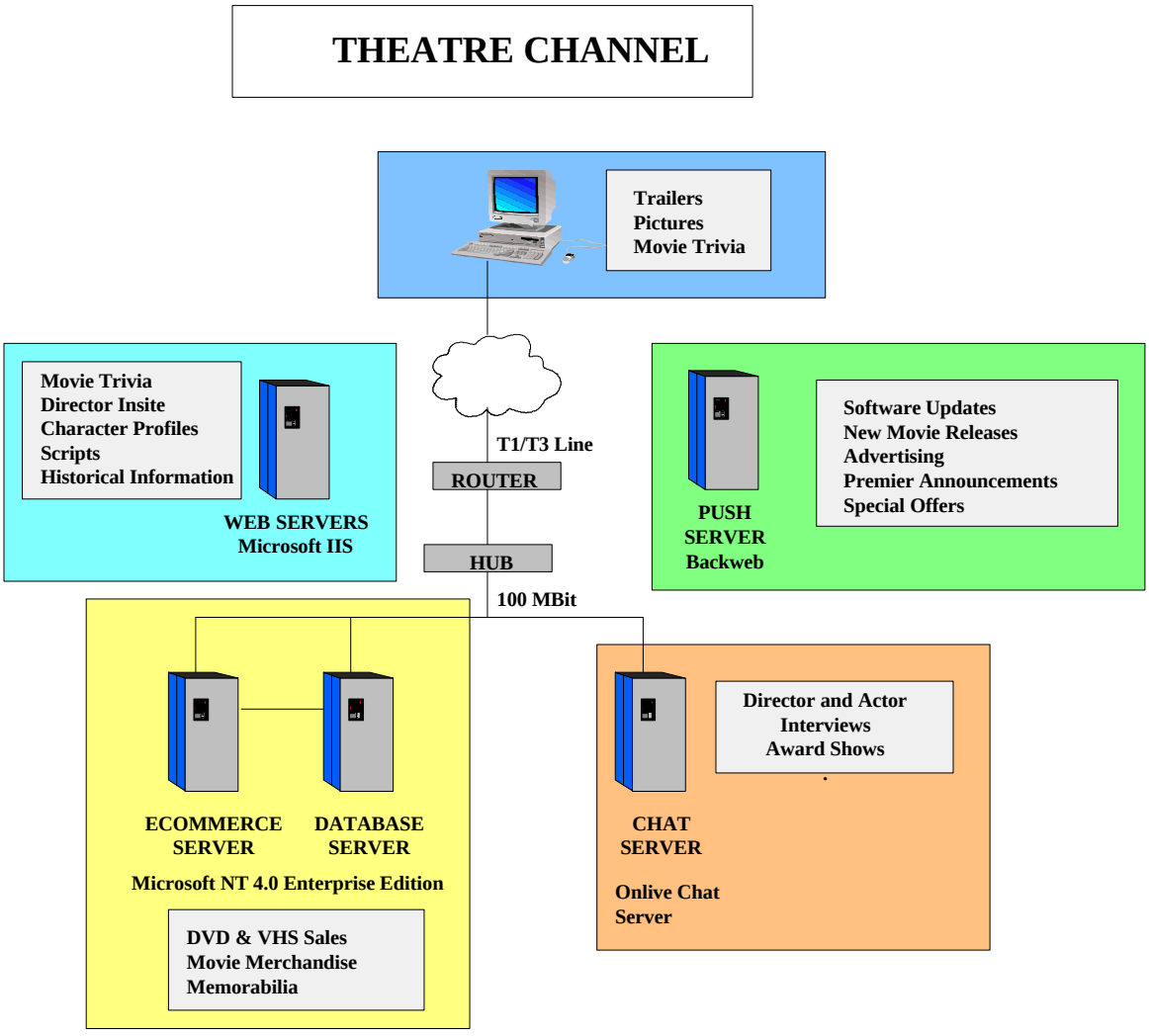

Figure 4-9 Theatre Server Architecture

Web Site

The Web site would contain information on new releases, directors, actors and background information on films. Online reviews by major movie critics could be stored on the Web site, and users could go online to learn interesting facts. Special effects, location and set information could all be available for users to download, and a movie could also provide material on characters and story lines as well as background or historical information. Character profiles could be expanded to give movie fans a better understanding of each character. These are just few possibilities for enhancing the movie channel application.

The other major online feature would be electronic commerce.

E-commerce

A big part of major movie releases is the promotion of merchandise. Blockbuster hits such *as Jurassic Park* and *Star Wars* make millions of dollars on sales of products related to the movie. *Star Wars* action figures, for example, could be sold on the Internet, and movies based on novels could help sell books online. Ad banners promote products and link fans to movie Web sites.

Live Interviews with Stars and Directors

Movie fans could bring up live interviews with movie actors or even directors, who could give insights into the production and shooting of the movie. Special effects artists could describe how certain special effects were created; costume designers could describe the different fashions and clothes used in the movie. Live interviews provide great opportunities for direct movie promotion.

Summary

A full-featured movie channel can add more value than just a simple Web site alone. The current versions of Microsoft's Internet Explorer* and Netscape Navigator* add the concept of channels. Updates to a Web site can be downloaded on the user's machine for off-line browsing. The theatre channel used Active Channels to download and keep updated the Web pages used in the movie application. The Link Resolver and Dynamic HTML was used to create a channel with richer multimedia. Movies are promoted best by delivering what makes them great in the first place—great-looking video. Anything short of full 24- or 30-FPS video degrades the user's experience. Integrating video stored on CD or DVD along with Dynamic HTML creates a more compelling theater channel.

Chapter 5 - Game Channel Application

The Game channel is an application that allows for the distribution of game advertising and previews. The application is strictly Web based and written entirely in HTML and VRML. The game world is a VRML-based arcade that provides game kiosks represented in a 3D environment. The user can navigate through the 3D arcade and then choose a particular game.

After selecting the game, the user is shown a game kiosk. The game advertising plays, giving the user a video preview. He or she can then press the play button to get a stored video clip of game play. If interested, he or she can select the banner and jump straight to the live game site to look up more information on the game or even purchase the game online.

The 3D VRML graphics and the use of 30-FPS full-motion video are intended to provide a better experience than just viewing normal 2D Web pages. The response to the user is faster, and the quality of audio and video is much better than can be delivered over the Internet.

WebCD* publisher is used to develop the game channel, which allows for the creation of CD-ROMs that mirror existing Web sites. The publisher provides the features necessary to manage and create a rich multimedia channel.

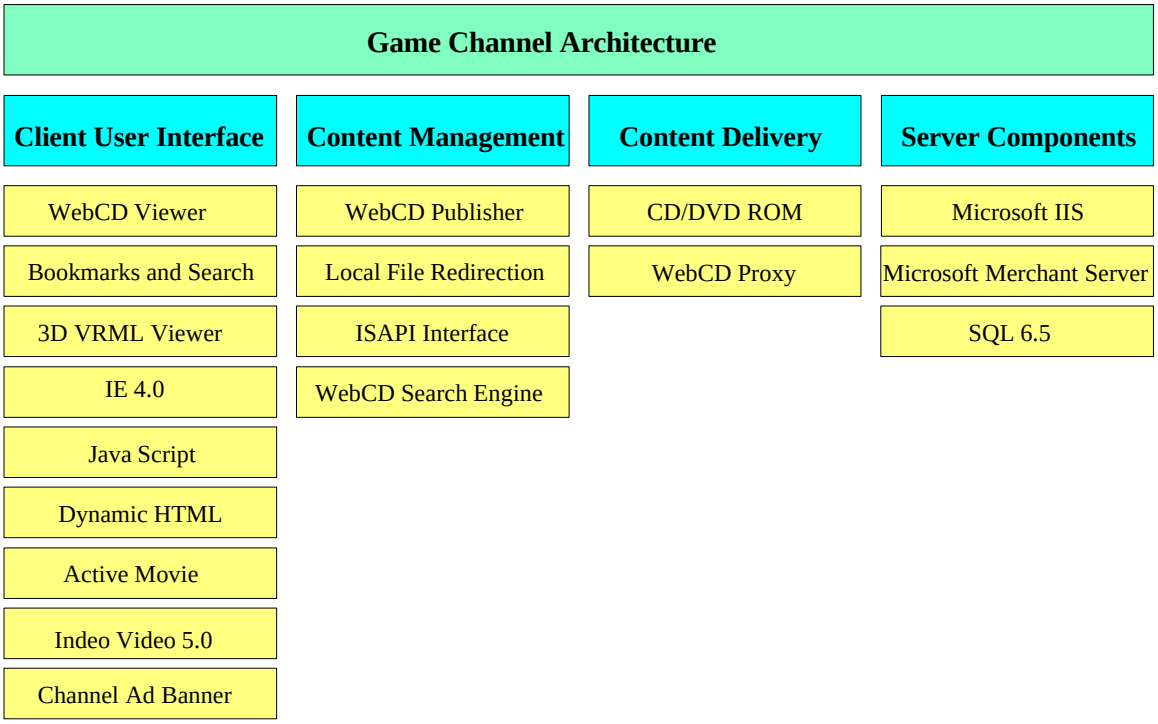

Figure 5-1 Game Channel Architecture

Client User Interface

The Game Channel interface is built around WebCD* Viewer, which optionally displays normal Web pages in a full screen kiosk mode, and adds a search engine and bookmark feature for easy navigation of large Web sites. A game arcade was developed using 3D studio Max* to demonstrate a 3D world into a channel; once an arcade game is selected, the user is presented with a game kiosk, created using dynamic HTML and Java Script. A ticker tape Java applet is included to provide advertising from multiple channels, and ActiveMovie* and Indeo Video 5.0 add rich multimedia to the application for game previews and advertising.

Figure 5-2 Client User Interface for Game Channel

Virtual Reality Modeling Language (VRML)

VRML is a language for describing 3D objects and possible user interactions with them. Using VRML, developers can build a sequence of visual images into Web pages that a user can interact with by moving, rotating and clicking within a 3D scene. The game channel uses a VRML-based arcade for users to move within and select the game they are

interested in. The VRML world was created using 3D Studio Max*. Anchors were placed on 3D objects to jump to a two dimensional HTML page displaying the game kiosk. All the VRML assets and scripts defining the world resided on the Web.

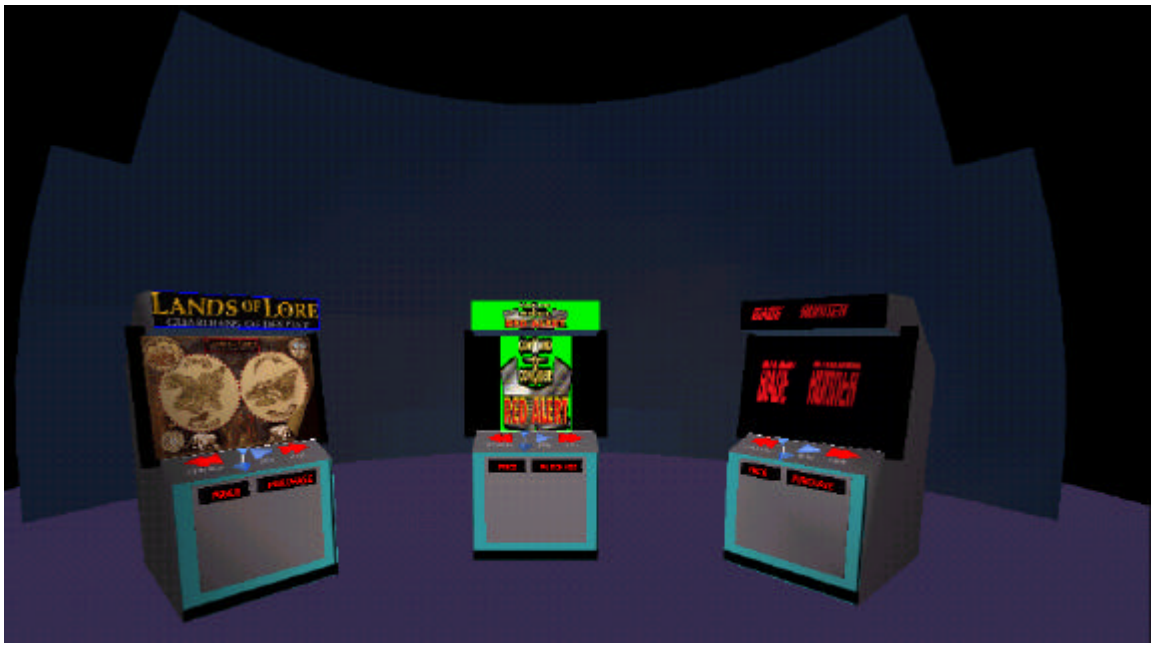

Figure 5-3: 3D VRML game world

Advertising Banners

The advertising banners in the game channel delivers ads from multiple sources. The banners are the one live component of the game application. The CD contains all assets except the Java applet and the ad banners. A directory for each channel is created to deliver advertising banners targeted at a specific channel audience. Data in each directory includes an image file named <name>.gif" to be displayed by the Ticker Tape. The GIF image files are named using consecutive number starting from 0 and appropriate extensions, in this case GIF (ie, 0.gif, 1.gif, 2.gif, …). For demonstration purposes, the ad-banners included game banners but could consolidate advertising from multiple channels. This applet could easily be used as a Webtop using Netscape's Netcaster* or a Microsoft Internet Explorer* active desktop to be a generic event notification vehicle for channels.

```
<APPLET code="TickerTape.class" codebase="..\Ticker"
align="baseline" style="position: absolute; top: 22; left:
224" width="352" height="72" id="BandBox">
<param name="speed" value="20">
<param name="dimension" value="600,64">
<param name="delay" value="3000">
<param name="Channel0" value="Lands of Lore,0,1,null">
 </applet>
```

Figure 5-4 Channel Advertising Java Applet

Embedding Video for WebCD* Publisher

The game application uses Indeo Video 5.0 to provide 30FPS video embedded within an HTML Web page. Standard HTML is used to embed the AVI video clip that will be redirected to the CD by WebCD* publisher. The code to embed the movie clip is shown below with the appropriate parameters such as height, width and position. Dynamic HTML is used to position the graphic precisely on the background image, and WebCD* publisher is used to retrieve the asset and then redirect it to the file stored on CD-ROM.

```
< IMG dynsrc= "HTTP://134.134.246.52/game/lol.avi"
  start="fileopen"
  width="320" height="240"
  style="position: absolute; top: 138; left: 238"
>
```

Figure 5-5 Embedding video clips into an HTML page

The next step is to create bookmarks to HTML pages stored on the CD-ROM or live on the Web. This is a simple process of marking each intended HTML page as a bookmark.

Kiosk Mode

Kiosk mode provides for a full-screen display of your WebCD application. The normal browser menu buttons and toolbars are no longer available. This mode allows for fully branded and customized interfaces for your applications. The home and starting pages are set before burning the CD.

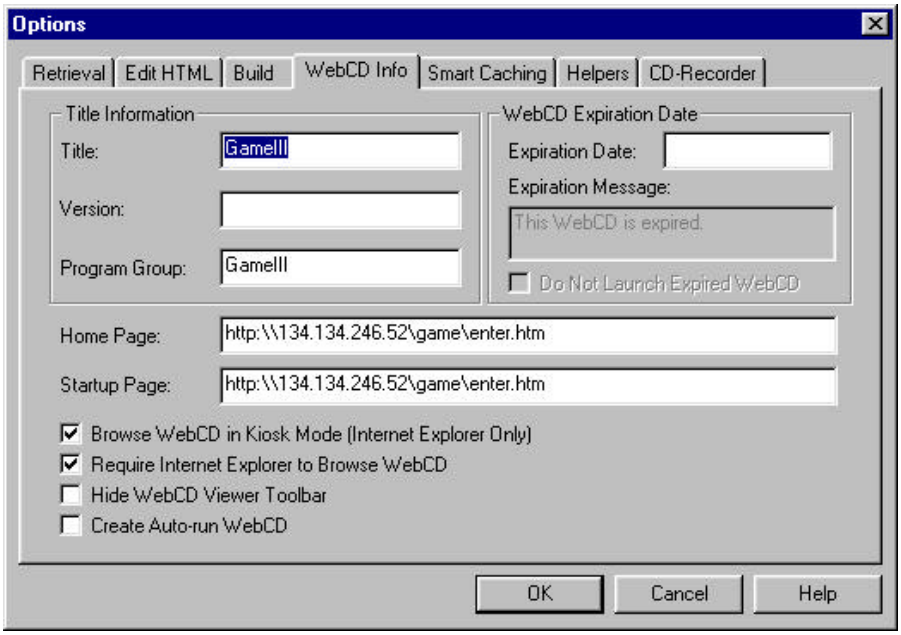

Figure 5-6: Kiosk Mode

Content Management

The process for creating a WebCD application consists of three phases: content retrieval, setting bookmarks and searches, and building the CD. Large Assets should be tagged for local redirection to CD-ROM.

Web Retrieval and Bookmarking

The first step in creating the WebCD* publisher application is to browse the site and select assets to be retrieved on the CD-ROM. It provides an automatic retrieval mechanism. You select the starting page, and the Web publisher will retrieve all the links within the starting page.

WebCD* publisher is arranged with a simple hierarchical view of the Web site on the left and a live browser on the right. Retrieved HTML documents and graphics are displayed on the left hand side. The developer surfs the Web site(s) on the right and then marks the URL for retrieval on the left. Assets can also be marked as not to be retrieved, in which case this content will be retrieved live from the Web site. An additional feature of WebCD* publisher is that it will mark bad links as you retrieve them. This ensures that the final product delivered on CD-ROM will not contain missing links. When the retrieval process is completed, the CD-ROM can be tested, ensuring that all pages were retrieved correctly.

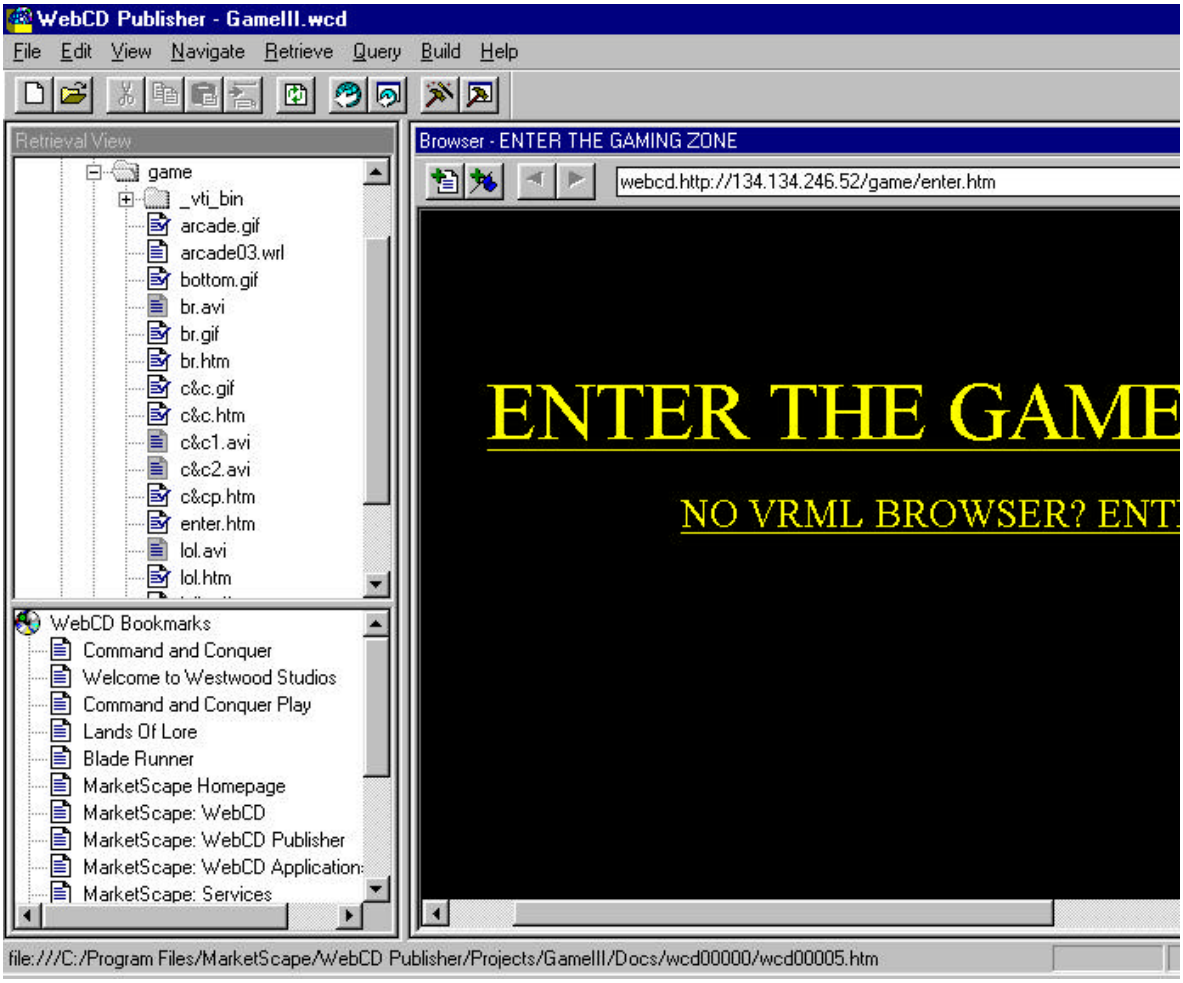

Figure 5-7: WebCD Publisher

File Redirection

File redirection is used to store the large Indeo Video 5.0 AVI clips. Pressing the right mouse button over the URL or asset and then selecting local redirection brings up the local redirection window. To prevent these assets from being cached, the files must be stored in the docs\wcdother directory on the CD image. The original URL is redirected to the CD-ROM file set in the local file text window. This simplifies the inclusion of rich multimedia assets into an existing Web site. Sizable assets can be created to provide higher resolution images, or better quality audio or video.

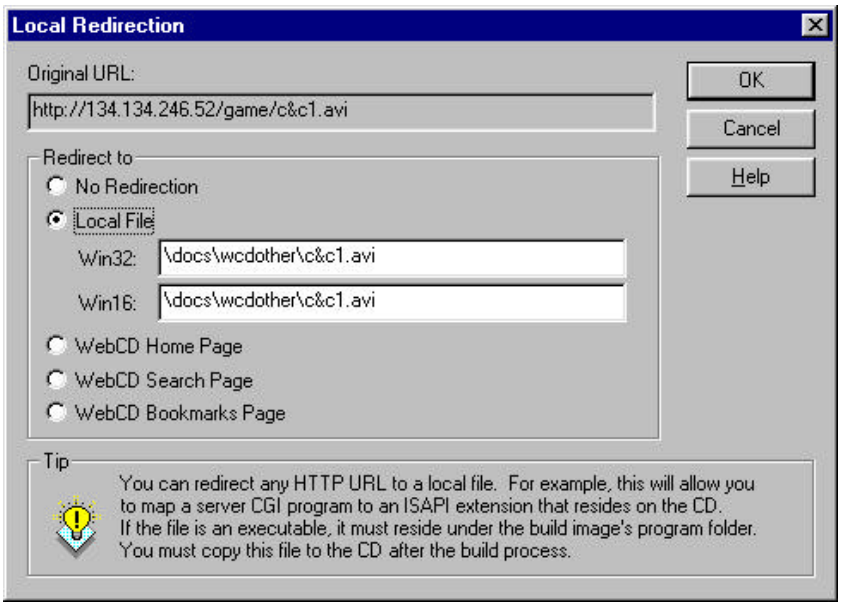

Figure 5-8: File Redirection of Large Assets

Content Delivery

After the Web site is retrieved using WebCD* publisher, a CD-ROM can be produced. This CD-ROM can be browsed off-line without the performance problems associated with the Internet. The greatest benefit is the addition of richer multimedia assets to enhance the standard Web site.

Content Updates

The CD-ROM is a direct image of what is stored on the Web Site. Application updates are accomplished by modifying the Web pages stored on the server. The WebCD proxy interface determines which assets are the newest and displays them within the viewer, and the user can browse the Web site off-line or connect to retrieve the latest updates. The game application mixed content stored on the CD-ROM with the latest advertising stored on the Web site.

Web Site Verification

To verify that your application will run on the Internet when live, you must set the URLs to your Web site(s). The URLs entered should validate that your server is running and that the user can connect live to your site. If the server is not running or the user is off-line, he or she may run in a faster mode that accesses only the CD. This is also where you set the cache setting for browsing the Web. Since our application uses both stored content on the CD and live content on the Web, the Refresher setting was used.

Summary

WebCD* Publisher is a useful tool for converting existing Web sites into rich and compelling channel applications. For large Web sites, it provides bookmarking and searching to provide easier access to key Web pages and information. The Web site can be made richer by adding higher resolution graphics, audio or video to the CD-ROM. When viewing the application, the user is accessing the Web site but gets better performance because the assets are local.

Chapter 6 – Summary

We've looked at three different Internet applications using a variety of technologies. Clearly, the Internet provides the possibility of delivering a large number of channels in the form of digital entertainment. Much like broadcasting, these channels compete for viewers among millions of Internet users.

As channels and Web sites evolve, broadcasters and the entertainment industry are becoming more and more involved with the development of content on the Internet. Currently, the most popular sites on the Internet consist of traditional forms of entertainment such as magazines, television, music, movies and news. Channels are enhancing the entertainment value of these products. Consumers expect equivalent quality to TV entertainment and will expect nothing less from channels delivered to the PC.

In order to create successful digital channels and entertainment, channels need to make use of real-time delivery of video and audio for live broadcasts. Too, advertising and electronic commerce are critical to make channels economically profitable. Channels require updated and compelling content to keep viewers connected, and can differentiate themselves by providing 3D graphics, animation, video and audio into an integrated, easy-to-use interface that ultimately provides the best possible user experience. In the three applications, we have touched on just a few of the technologies used to enhance a channel. Refer to the appendices for other technologies and data types which can be used to build a compelling channel.

The Pentium® II Processor based computer is the ideal delivery vehicle for these emerging digital channels. The PC's flexibility in handling a wide variety of rich data types, including the communication infrastructure to deliver compelling content, makes it the platform of choice. The integration of these data types into a user interface that attracts users is the challenge of the developer, who should aim high or risk losing out to digital channels that do.

Ultimately, the digital channels with the most compelling content delivered in a rich manner will capture the largest consumer audience.

Appendix A – Audio and Video Codecs

A wide range of codecs are available for delivering rich multimedia to the desktop, but not all possible codecs are listed in this section. Which codec to use depends on application requirements and bandwidth limitations. Storage requirements are also a consideration, because richer multimedia require higher bandwidths. High-quality video codecs such as Indeo Video 5.0 and MPEG1 deliver good-quality video at about 1.2 Mbps of bandwidth. Use of MPEG1 and Indeo Video 5.0 may be good for applications delivered on CD-ROM, but great-quality video can be obtained by using MPEG2 video. This compression algorithm provides broadcast-quality video at about 4 Mbps. Although it requires a large amount of storage, it is the codec of choice for DVD-ROM.

Codec	**Size**	**Size/1 Minute Video**	**Download Time 1 Min/28.8 Modem**
Raw 640x480 (30 FPS)	221.184 Mbps	1.6 Gigabytes	128 hours
MPEG4	28.8–64 Kbps	0.17–0.48 Mbytes	1–2.2 min.
H.261	64 Kbps–2 Mbps	0.48–15 Mbytes	2.2–69 min.
Indeo Video Interactive	50 Kbps–0.5 Mbps	0.37–3.7 Mbytes	2–20 min.
MPEG1	1.2–2 Mbps	9–15 Mbytes	41–69 min.
MPEG2	4–60 Mbps	30–450 Mbytes	138–2,088 min.
Motion JPEG	10–20 Mbps	45 Mbytes	-694 Nub

Table A-1. Rich multimedia assets storage and downloading requirements

The type of compression used by the application will determine how many assets can be stored on the CD. Table B-2 lists some of the possible audio and video codecs and how many minutes of the assets could be stored on a CD- or DVD-ROM. The large storage capacity of CD- and DVD-ROM make them much better for delivering rich multimedia assets.

Asset	**Size**	**CD-ROM (648 MB)**	**DVD (4.7 GB)**
MPEG1	1.2 Mbps	72 min.	522 min.
MPEG2	4 Mbps	21.6 min.	156 min.
Indeo Audio	16Kbps-176kbs	496min-5456 min	3,854 – 39,424 min
44-KHz audio	1.4 Mbps	62 min.	448 min.
Real Audio* (28.8	28 Kbps	3,000 min.	21,759 min.

Kbps)			
AC3* (5.1 Channel)	383 Kbps	225 min.	1,636 min.

Table A-2. Storage capacity of CD- and DVD-ROM

Video Codecs

Intel Indeo® Video 5.0 Progressive Download

Indeo Video 5.0 compression and playback software allows for high-quality, scaleable video content to be viewed over the Internet or intranet quickly and easily. Its progressive download technology allows you to start seeing video as it downloads. Users have the option of continuing to view the video while determining whether to save it on a hard disk. Indeo Video 5.0 allows you to create high-quality video for Web pages and be assured that users can experience your video at the best level of quality that the connection speed will allow. Indeo Video provides the best compromise between good quality and scaleable bandwidth.

Indeo Video 5.0 is a good codec for delivering high-quality video on CD-ROM as well. The three sample channels were created using Indeo Video 5.0. The codec is ideal for channels requiring video on demand on the Internet as well as video stored on CD- or DVD-ROM.

MPEG2

Rich video should consist of a full-screen experience that is a minimum of 24 frames per second (FPS). For broadcast quality, MPEG2 is the primary choice. It provides good-quality video and is typically used in satellite broadcasting or for distribution on DVD-ROM. Software playback of 24-FPS MPEG2 requires a minimum of a 266-MHz Pentium® II Processor based system with hardware motion compensation on the graphics card. This setup provides the best-quality video for the consumer. With a bandwidth ranging from 2–8 Mbps, pushing it across the Internet is not practical across 28.8-Kbps modems. For broadband, MPEG2 is a viable choice and could be delivered to the end user across satellite, cable and possibly ADSL modems.

MPEG1

MPEG1 is a viable alternative for delivering VHS quality video on CD-ROM. Requiring from 1–2 Mbps, the codec requires slightly less bandwidth than MPEG2. The quality of the video is not as good but can be used to provide video for distribution on CD-ROM. Microsoft's Netshow* and Xing's Streamworks* provide real-time video streaming servers for real-time broadcast across a network. The high-bandwidth requirement limits its use to private intranets, networks, or for possible use in broadband delivery.

MPEG4

For delivery across 28.8-Kbps modems, a variety of codecs can be streamed across the Internet. MPEG4 is designed to be delivered across 28.8-Kbps modems. The video quality is acceptable and provides a frame rate on the order of 4–8 FPS. MPEG4 is also scaleable up to 56 Kbps and is a relatively good solution for delivering low-end video.

H.261

H.261 is a good low-end codec that is scaleable across a variety of bandwidths. The algorithm is typically used for video conferencing and can provide a good solution for live video broadcasts across the Internet or private networks. It is highly scaleable, and for bandwidths from 64–400 Kbps, it provides reasonable video quality and frame rate.

Summary

Which codec you choose depends on the application, and whether the video is being delivered in real time. For channels, the best possible choice is MPEG2. Video can be distributed on CD- or DVD-ROM to provide the end user with the best-quality video. Preserving the broadcast quality of movies and music videos are important for the user's experience.

Audio Codecs

A variety of streaming audio codecs provides audio across the Internet. These codecs can also be used to store the audio. Streaming audio can be used in channels to provide live concerts, interviews with artists, presentations and music clips. Creating a streaming audio codec requires an audio server and an audio source. Typically, an audio codec requires a Microsoft Windows NT* Server and a standard audio card to deliver streamed audio on the Internet. Most audio codecs require the client software to be loaded on the user's machine.

Intel Indeo audio software

Intel Indeo audio provides high-quality compressed audio for Internet, intranet, and multimedia applications. It can compress both music and voice audio as much as 8:1 with no appreciable loss in quality. Intel Indeo audio works in conjunction with Indeo Video 5.0's Progressive Download capability to allow richer, fuller, multimedia Web sites.
The Indeo audio codec comes included with the Indeo video 5.0 codec and progressive download filter. Movies containing compressed Indeo video 5.0 and Indeo audio streams are created using Intel's Indeo video 5.0 Progressive Download Publisher tool.

Real Audio* Server

Real Audio* consists of a Real Audio player, Real Audio server and Real Audio encoder for delivering real-time audio across the Internet. The Real Audio* encoder feeds the server for delivery of the stream across the network. Real Audio uses HTTP streaming for delivery of audio across the Internet. Real Audio can be delivered live or stored in a file. You can download the player for free from the Real Audio site.
Real Audio* provides a scaleable audio solution that can deliver a variety of audio streams. Audio quality ranges from AM to close to CD-quality sound. At 28.8 Kbps of bandwidth, the audio is close to FM quality. The Real Audio server is capable of delivering either UDP or TCP packets to the user. UDP does not guarantee the delivery of packets across the network. UDP is often used in real-time streaming, since there is no need to retransmit when the content is in real time. The real-time audio server can be set

up to deliver real-time streaming audio from 28.8-Kbps to ISDN rates, depending on application requirements.

Codec	Bit Rate	Size/1 Minute	Download Time 1min/28.8 Kbps	Application
Wave Audio (44-KHz Stereo)	1.4 Mbps	10.6 Mbytes	49 min.	Music
Dolby AC3 (5.1 Channel Audio)	383–448 Kbps	2.8–3.64 Mbytes	13.3 min.	Music
Voxware*	1.3 Kbps	9.750 Kbytes	0.05 min.	Voice
Indeo audio software	16 –176 Kbps	0.12–1.32 Mbytes	.55-6.1– min.	Music, voice
Real Audio* Dual ISDN	128 Kbps	0.96 Mbytes	4.4 min.	Music
Real Audio* 3.0 (28.8 mono)	28.8 Kbps	117 Kbytes	1 min.	Music, voice
Real Audio* (4.0 8.5 Kbps)	8.5 Kbps	64 Kbytes	0.30 min.	Voice
Xing StreamWorks* LBR	8–16 Kbps	60–120 Kbytes	0.27-0.55 min.	Voice
Xing StreamWorks* MPEG1	32–192 Kbps	0.240–1.44 Mbytes	1.10-6.67 min.	Voice, music
Xing StreamWorks* MPEG2	8–80 Kbps	60–600 Kbytes	0.27-2.7 min.	Voice, music
Liquid Audio*	28.8 Kbps	117 Kbytes	1 min.	Music
True Speech*	8.5 Kbps	63,750 Kbytes	0.30 min.	Voice

Table A-3. Audio Codecs

XING StreamWorks*

Xing provides StreamWorks* products for delivery of MPEG1 and MPEG2 audio. The standard audio codecs provide scalability depending on the application. LBR can be scaled from 8–16 Kbps, MPEG2 can be scaled at 8–80 Kbps, and MPEG1 can be scaled from 32–192 Kbps. The ability to scale allows the audio to be compressed according to the fidelity requirements of the content. This maximizes the efficient use of bandwidth across the Internet.

The Xing StreamWorks* solution also includes a player, server and transmitter for delivering streaming audio or video. The Xing StreamWorks* server delivers live and stored audio over the Internet, and enables real-time transmission of MPEG audio and video using either multicast or unicast. This provides a scaleable solution for delivering low-bandwidth MPEG1 or MPEG2 audio and video up to high-quality MPEG 1 video. The standard implementation is one of the alternatives for delivering scaleable audio and video across the Internet.

Liquid Audio

Liquid Audio* provides a comprehensive solution for delivering online music to end users. It provides downloadable audio based on Dolby's* digital compression technology. The

solution includes a music server to store and stream real-time music to the end user, and the liquid music player can also download lyrics and artwork stored on the Web for use on a CD-album or in a piece of music. Electronic commerce is built so that music can be purchased online using credit cards. Liquid Audio provides a viable solution for electronic delivery of online music.

True Speech*

True Speech* is a relatively good-quality codec running at 8.5 Kbps. For voice applications, the True Speech* algorithm is useful for delivering low-bandwidth audio across the Internet in real time or as stored content. It could possibly be used in streaming audio applications that require voice only.

3D RSX

3D Realistic Sound Experience positions sounds in 3D space. Sounds can be created as if they are in front, behind, left or right in 3D space. Application sound can be even more realistic to the end user as he or she interacts with the channel. Sound effects such as reverberations and Doppler effects can give the sensation of cars approaching and then receding. This form of 3D audio animation gives the user the sensation of getting hit by sounds from various directions. 3D RSX can be incorporated into VRML 2.0 worlds or streamed across the Internet. By giving the user the best possible audio experience and a sense of direction when navigating 3D worlds, 3D RSX is a good alternative to 2D sound.

Dolby AC3*

Most DVD movies digitized today use Dolby AC3* audio, which provides quality audio with a 12:1 compression ratio. Its high-quality compression makes it a viable codec for delivering a great audio experience with good compression. It is the standard choice for delivering video and audio on DVD-ROM.

Summary

The choice of audio codecs is dependent on the application. For live broadcast, a variety of solutions are available: Real Audio*, Xing StreamWorks*, and Liquid Audio* are just a few of the streaming solutions on the Internet. For audio storage, WAV and PCM audio are alternatives since they provide great quality. WAV audio was used in the music channel and could be delivered without compression across the net using push technology. It is also ideal for delivery on CD- and DVD-ROM. If a large number of music or audio samples are required, Real Audio*, MPEG1, MPEG2, MPEG3 and liquid audio* are alternatives.

Microsoft Netshow 2.0*

Microsoft Netshow* provides a bandwidth scaleable solution for delivering streaming video, audio or data. It provides the ability to multicast or unicast streams on the Internet. The Netshow* 2.0 product line can synchronize the delivery of audio and graphics in the form of live presentations. Audio streams can be delivered in real time with synchronized Web surfing. Netshow allows the embedding of URL references in the stream, so Web pages can be pulled up while the audio is playing. For example, a live concert could be

delivered while information about the band or advertising is pulled up at the same time through Web pages.

The Netshow *2.0 channel server consists of administrative tools for managing programs and channels. A single Netshow channel server can support multiple streams and deliver several streams of programming on the Internet. The Netshow server broadcasts events on the Internet using an ASX file. The simple file format specifies the program's ASF file as well as the server that will deliver the programming. The Netshow administrative tools consist of a channel and program manager to manage and deliver live audio and video programming across the Internet.

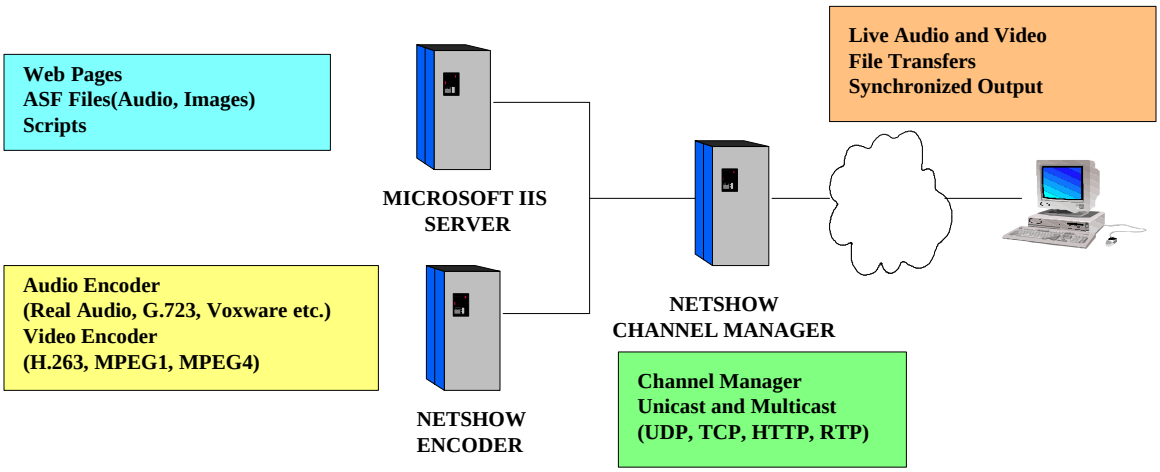

Figure A-4. Netshow architecture

The client software consists of a Netshow player, which is incorporated as an option in Internet Explorer 4.0. A Netshow file can be embedded in a Web page or can run as a stand alone application. Once the client is loaded on the user's machine, he or she can view Netshow broadcasts.

Netshow* Server

The Netshow* server provides the capability of delivering audio, video and data programming on the Internet. Netshow 2.0 provides the administrative tools to define digital channels that can be scheduled for delivery using unicast or multicast. Acting much like a broadcast of TV channels, it allows for setting up channels that can deliver a variety of content scheduled at specific times.

Multiple channels can be delivered and broadcast 24 hours a day. The program manager defines the streams to be delivered, and the channel manager determines the how the stream is delivered on the Internet.

Announcements can be delivered to the end user describing a future programming broadcast that may match his or her programming interests. If the user is interested in the broadcast, he or she can select the program and watch it live. Broadcasts can also be stored in a file for delivery on demand.

Netshow* Player

The Netshow player is capable of running within a Web page or stand alone. In stand-alone mode, the user can watch a program of streamed audio, graphics or video. The user can watch the digital program without interaction, or interactive features can be included. The Netshow player is integrated with the latest version of Internet Explorer. Live events can be delivered to the user's desktop as part of Internet Explorer's active desktop.

ASF Format

The ASF format provides a method for synchronizing different data types and providing error correction for delivering audio and graphics across the Internet. Microsoft provides the Netshow* ASF editor, which allows for a multimedia show comprising the integration of video, audio and text. The ASF editor manages the bandwidth so that delivery of the different multimedia content fits within the expected bandwidth. The content is scaleable and can be set up to deliver content to modems, ISDN or intranet connections. With more bandwidth, better-quality video, audio and graphics can be delivered on a channel.
The Netshow* server can provide several ASFs created for different bandwidths. This gives the user a scaleable experience depending on network connection. Mixing video, audio and data provides a richer experience than any of them alone.

ASX Files

ASX files are generated to announce programs to clients. An ASX file is a basic file declaring the protocol being used, the name of the server and the ASF file being used. It redirects the user to the appropriate streaming servers, and can be included on any Web server and incorporated into any HTML reference.

File Broadcasting

The Netshow* 2.0 server is also capable of delivering files to the user. It can provide software updates to a large number of users with the same broadcast paradigm as delivering audio and video.

Netshow* Broadcasting

The Netshow* server allows for multiple channels to be set up and delivered. Multiple channels can be created to contain live broadcasts of video or audio. Stored content can also be delivered to the user and contain video, audio or data synchronized using ASF format. Much like TV broadcasting, Netshow channels deliver content at scheduled times.

Netshow Video and Audio Servers

Netshow's* architecture provides for connecting several live servers to Netshow servers that deliver the streams to connected clients. The Netshow solution is highly scaleable and can deliver multiple digital broadcasts to a large number of users. According to Microsoft, a single Netshow server is capable of supporting up to 1,000 streams at 28.8 Kbps.

Support for Multiple Codecs

Microsoft Netshow* 2.0 supports a variety of audio and video codecs. Table A-5 lists some of the supported codecs. At 28.8 Kbps, reasonable-quality audio and video can be

delivered to the user with MPEG4, Real Audio and H.263, but streaming audio provides a much better real-time experience.

Codec	Bandwidth	Application
Voxware Metaware*	2.4 Kbps	Voice
Lernout and Hauspie*	4.8 Kbps	Voice
Real Audio*	Scaleable (28.8 & ISDN 128 Kbps)	Music
Fhg* MPEG Layer 3	8 Kbps	Music/voice
MPEG4	28.8 Kbps	
Vivo* G.723 and H.263	28.8 Kbps	Good-quality 4–8 FPS video
Intel G.723 and H.263	28.8 Kbps	Good-quality 4–8 FPS video
MPEG1	1.2–2.0 Mbps	VHS-quality video

Table A-5. Supported codecs in Microsoft Netshow* 2.0

Appendix B – Content Development Tools

Java

Java has become a popular programming solution for delivering network applications, providing an object-oriented paradigm for client-server applications. Using Java, Web pages can be programmed to add animation and special effects to objects with simple scripts. The benefits of Java in developing channels is in the use of components written as Java applets. (Applets handle special effects, animation, ad banners, text formatting, 3D effects, databases and so on.) Since Java is not platform-dependent, it is considered a good programming environment for delivering client/server applications. Its support in both Netscape 4.0 and Microsoft Internet Explorer*-- makes it a good choice for developers, as well. Java applets within Web pages to enhance their look and feel.

Another benefit to Java applets is their use as Webtops or active desktop components. Simple applets can be created to provide features such as headlines, notifications, banners and tickers. In game applications (see Chapter 5) a Java-based applet was designed to deliver advertising banners from multiple channels.

The applet is not very CPU intensive, but when integrated with rich multimedia such as Indeo Video 5.0, the applet tends to run into performance problems on low-end machines. The integration of Java applets and rich multimedia content requires high performance systems in many cases.

*Microsoft ActiveX Technology**

Much like Java applets, ActiveX controls are intended to be reusable building blocks for applications. A variety of ActiveX components provide functionality similar to Java applets. Several ActiveX controls were used in the creation of several of the channel applications. An URL Link Resolver was used in the theater application to retrieve stored AVI clips off the CD-ROM. Microsoft Wallet's* Address and CreditCard ActiveX controls were used to capture address and credit card information. The developer can choose from a wide variety of both ActiveX controls and Java applets building blocks to develop channels.

Dynamic HTML

Dynamic HTML and the ability of both Netscape's Netcaster* and Microsoft's Internet Explorer* to handle HTML elements as objects makes livening up a Web page much easier. Scripts can be written to make elements animate across the screen. Special effects can be used in text and graphics to add professional-looking graphics and typesetting to a Web page. Microsoft's Dynamic HTML features "scriplets", which allow for distribution of HTML and code as objects residing anywhere on the Internet.

Software Agents

Software agents are interactive personalities that can act as guides or companions for channel users. These agents can be added to provide an interactive, conversational interface that leverages the natural aspects of human social communication.
Microsoft provides several agent application programming interfaces (APIs) that make agents accessible to programming languages that support ActiveX technologies. For more information regarding Microsoft's agent, visit www.microsoft.com/workshop/prog/agent.

VRML

VRML 2.0 is an emerging standard for providing 3D virtual reality worlds across the Internet. VRML consists of a text description of a 3D world. The text contains nodes that form the basis of VRML worlds. These nodes contain information about 3D objects, animations, lights, colors, sounds and video. Four basic primitives are put together to define a VRML world: box, cone, cylinder and sphere. These building blocks can be combined to form a 3D world that a user can navigate through and experience in real time. VRML combined with 3D sound can create a very interactive virtual world.
A lot of the 3D worlds on the Internet are used to generate a community in which users can gather and interact with each other. Included in these worlds are avatars that interact with individuals. Advertising avatars can be deployed to interact with users and act as virtual salesmen. Avatars with information about a variety of products can help promote electronic sales on the Web.
VRML worlds require large multimedia assets, such as textures, that need to be downloaded. Depending on the 3D world, they are sometimes too large to download over a 28.8-Kbps modem. To get around this problem, the textures and entire VRML world could be distributed on CD. For our game channel, the 3D VRML world is distributed on the CD to provide real-time performance when accessing the world.

Active Server Pages

Active Server Pages (ASPs) allow for dynamic content creation within a Web site-based on scripts written on the server side. This is in contrast to Java and Visual Basic scripts run scripts on the client side. ASPs are designed to provide a powerful scripting language for delivering dynamic Web pages to the user. The output of ASPs return standard HTML pages after running scripts on the server side. This flexibility is useful in delivering content based on user preferences or even asset resolution on the client's PC. The Microsoft Merchant Server* is written entirely in ASPs and is easily customizable.

Appendix C - Client User Interfaces and Internet Browsers

Client-Side User Interface

A *client* is the requesting program or the user in a client/server relationship. For example, the user of a Web browser is effectively making client requests for pages from Web servers. Client user interfaces (CUIs) are extremely important, because they provide the point at which the user accesses data and interacts within the application. In the world of channel architecture, a few characteristics define a CUI:

- Full screen
- User customizable and manageable
- Seamless integration of push technologies and data delivery notification mechanisms

A full-screen UI allows the user to view active channel content from the selected channel. This provides a richer user experience, in that the delivered content does not compete with other content available on the PC. A full-screen UI also allows you to remove other brand names from the experience. In the past, when users accessed the Web, they probably used an off-the-shelf Web browser. This browser would typically be branded with the commercial software provider's name and brand, which may detract from the value of the channel.

User customization and manageability go beyond the client interface. The user should be allowed to dictate the manner in which data is delivered and stored on the client PC. The CUI should provide an appropriate way to perform this function.

As new channel content is delivered to the client PC, the user needs to be notified of its arrival, especially if it is time sensitive. For example, if your channel has an e-commerce feature and you push notification of an item going on sale for only 12 hours, you want to get the user's attention as soon as possible. Several issues need to be addressed in this case. First, you don't know what the user will be doing when the update is received. Therefore, your notification mechanism must be passive so that it doesn't interrupt the user's current activity. Two examples of some unobtrusive notification mechanisms are a scrolling marquee text field across the top or bottom of the screen, or flashing icons in the System Tray of the Windows 95 Taskbar.

Several commercial products and technologies offer CUIs that adhere to the channel architecture. Microsoft's Internet Explorer* and Netscape Communicator* 4.0 are the two major browsers that provide a user interface for the development of Internet applications. A wide variety of tools, plug-ins and controls can be used to create compelling channel applications based around the two browser solutions. In addition, Macromedia Director provides a tool for creating channels.

*Macromedia Director**

Macromedia's Director* is an authoring tool for multimedia projects. Director provides the experienced user the ability to integrate several media assets into a common user interface. The end result with your project is an interactive movie that is played back on the PC.

Cast Members and Sprites

Director refers to the elements of your project as Cast Members – everything you see or hear is considered a cast member. This includes all text, bitmaps, sounds, buttons, digital videos, etc.
Sprites are objects that represent how cast members appear in your script. Your Director project essentially defines the appearance and characteristics of sprites.

Director Xtras*

Xtras are software components that extend the functionality of Director. As your movie needs to access Internet or play rich media assets that are not natively supported by Director you can use an Xtra that has been developed to facilitate that function. Third-party developers who want to exploit special features and options typically supply Xtras.

Shockwave*

Macromedia has developed several media players that work in conjunction with their different development environments. These players are referred to as Shockwave players. These players must be installed on the client to enable the features within the content.

*Netscape Communicator 4.0**

Netscape Communicator 4.0* is one of the most popular browsers on the Internet. The release of 4.0 has incorporated dynamic HTML as well as Netscape's Netcaster for delivering channels to the desktop. Dynamic HTML in Netscape contains style sheets for formatting Web pages, downloadable fonts and HTML object positioning. With scripting, the Web developer has more control over the presentation of Web pages, including adding special effects and animation. Channels are implemented using Netcaster, which uses Marimba's Castanet to deliver Web pages and application updates to the user.

Netcaster

Netscape Netcaster* is a component of Netscape Communicator* 4.0 that fits into the channel architecture. Netcaster provides the ability to define channels that include features such as scheduling updates, locally cached content and personalized content. "Netcasting" is the process of pushing channel content onto your PC—making it available for the CUI. Netcaster works with standard Web servers to deliver the push content to the CUI. Extended support has been implemented to integrate the Marimba Castanet Tuner* to provide a dedicated push server.

Webtops

Netcasting is combined with another technology called Webtops. Webtops are intended to be cross-platform and run on a variety of operating systems. Webtops are HTML-based

interfaces that can be anchored to your desktop and allow content to be displayed based on the user's preferences. Webtops can be displayed full screen and provide branded CUIs. They can provide smaller, windowed interfaces on the desktop for providing real-time information or announcements. Webtops allow the developer to create a customized user interface for displaying content on the Internet.

Channel Finder

The Netscape Channel Finder is the remote control that allows channels to be added, removed and configured. Netscape provides a central location for subscribing to channels online.

Marimba Castanet

Marimba Castanet* provides the push solution for Netscape's Netcaster. Castanet is able to deliver Web pages or applications over the Internet and keep them updated on the user's machine. Castanet can deliver updates efficiently, since only incremental changes are sent when updating an application. Castanet can deliver customizable content based on user feedback or preferences.

Microsoft Internet Explorer

Microsoft Internet Explorer* provides several new features that are useful for developing channels. They include dynamic HTML, active channels and the Channel Definition Format (CDF). Internet Explorer was used to help develop the theater application as an example of using these features in the simple development of a channel.

Active Channels

Active channels provide the ability to subscribe to a number of channels written for Internet Explorer. Users can set their preferences based on what channels they want to subscribe to, the time for downloading and the priority of the channel. Active channels manage favorite sites and channel subscriptions using smart pull.

Web Casting

Web casting in Internet Explorer allows users to subscribe to channels and automatically download Web pages. Downloads can be scheduled periodically at off hours. Content delivered from these sites can be customized using CDF files. These downloaded Web sites can be browsed off-line

CDF provides the framework for describing Web content to be delivered to the desktop. The CDF file defines the content to be delivered to the end user, how often it is updated and when the content is outdated. Several tools, including Microsoft FrontPage97*, are available to help define and manage the creation of channels.

XML – Extensible markup language

Our mention of CDF warrants a discussion about XML (Extensible Markup Language). CDF is a markup language derived from XML.
XML is designed to develop and extend new markup in a standard way. XML is designed to specify precise rules for how XML markup is formed so development tools and browsers will already know how to parse it. HTML is all about what content looks like, and XML is about describing the content so it can be manipulated online.
A typical XML document will consist of a prolog and content marked up with user defined tags (formed using XML rules). The prolog specifies the vocabulary (valid tags) and the grammar (hierarchical structure) of the XML tags in the document. The content is then marked up with the tags as you would do with HTML tags.
An XML parser could then use the tag specification and rules in the prolog to extract content from the document and do with it what it wishes.
CDF is a new XML vocabulary which Internet Explorer knows how to parse. IE takes the information in a CDF document and creates a client side channel.

Dynamic Hypertext Markup Language (DHTML)

Dynamic HTML is a newcomer to the channel architecture arena. A standard is currently being developed by the World Wide Web Consortium (W3C), and it is hoped that HTML 4.0 will contain formal specifications for this language.
The features that constitute dynamic HTML are included in Netscape Communicator* 4.0 and by Microsoft's browser, Internet Explorer*. However, each company has included a different implementation of dynamic HTML—the two browsers do not support a common feature set. Content developers will have to write different Web pages to support the lack of industry standards. Both Netscape and Microsoft have submitted their versions as proposals to W3C.

DHTML Concepts and Features

DHTML is a version of HTML that allows a Web page to change itself in place without a round-trip request to a Web server. This technology significantly decreases the time users have to wait for Web pages to be requested from the server and loaded on the client. Static HTML requires a server request to scroll through a list of items, but DHTML can control the entire process from the client machine.
DHTML provides an object-oriented view of a Web page and its elements, such as cascading style sheets and the layering of content, scripting or programming that can address all or most page elements and dynamic fonts.

Different Companies…Different Implementations

Microsoft and Netscape have chosen different implementations of Dynamic HTML. This has caused some confusion within the industry as the implementations have both commonalties and differences.
The features of DHTML that are supported by both Microsoft and Netscape include Object-Oriented Modeling, Cascading Style Sheets, Scripting Languages and Event Models.

An Object-Oriented View of Page Elements

Each page element (division or section, heading, paragraph, image, list and so forth) is viewed as a *hierarchy of objects/elements*. For example, each heading on a page can be named, given attributes of text style and color, and addressed by name in a small program or "script" included on the page. This heading or any other element on the page can be changed as the result of a specified event: a mouse passing over or being clicked, time elapsing, or "dragging and dropping" the image object with the mouse. (These event possibilities can be viewed as the reaction capabilities of the element or object.) Any change takes place immediately since all variations of all elements or objects have been sent as part of the same page from the Web server that sent the page. Thus, variations can be thought of as different properties of the object.

Not only can element variations change text wording or color, but also everything contained within a heading object can be replaced with new content that includes different or additional HTML as well as different text. Microsoft calls this "Text Range technology."

Style Sheets and Layering

A style sheet describes the default style characteristics (including the page layout and font type style and size for text elements such as headings and body text) of a document or portion of a document. For Web pages, a style sheet also describes the default background color or image, hypertext link colors and possibly the content of the page. Style sheets help ensure consistency across all or a group of pages in a document or a Web site. Dynamic HTML includes the capability to specify style sheets in a "cascading" fashion (that is, linking to or specifying different style sheets or style statements with predefined levels of precedence within the same or a set of related pages). As the result of user interaction, a new style sheet can be made applicable and result in a change of appearance of the Web page. You may have multiple layers of style sheets within a page, a style sheet within a style sheet within a style sheet, etc. A new style sheet may only vary one element from the style sheet above it.

Layering is the use of alternate style sheets or other approaches to vary the content of a page by providing content layers that can overlay (and replace or superimpose on) existing content sections. Layers can be programmed to appear as part of a timed presentation or as the result of user interaction. In Internet Explorer* 4.0, Microsoft implements layers through style sheets. Netscape supports the style sheet approach but also offers a new HTML <LAYER>...</LAYER> tag set (Microsoft does not support this tag set). The W3C Working Committee is considering both approaches, and both companies will reportedly support what W3C determines is the recommended approach.

Scripting or Programming

Although JavaScript*, Java applets, and ActiveX controls were present in previous levels of Web pages, dynamic HTML implies an increased amount of programming in Web pages, since more elements of a page can be addressed by a script or a program.

Microsoft's DHTML Subset

Microsoft has added a few features to its implementation of DHTML. These include several multimedia effects (Full-Screen Mode, Active Desktop and Dynamic Postitioning), Data Binding and Scriplets.

Full-Screen Mode

The Web browser can be viewed in full-screen mode. Branding can be applied to the Web page to provide a custom look and feel, including navigation buttons. Channels can be created that eliminate the browser interface and can prevent the user from easily exiting the channel. Links and the interface can be controlled to keep the user restricted to navigating within the application.

Active Desktop

Custom interfaces can be designed as well as active desktop components that can be used to broadcast announcements, live concert information, advertising specials or news items. These components could consist of Web pages or Java applets that provide a desktop notification mechanism for announcing new channel content or live events.

Dynamic Positioning

Objects such as graphics and text can be positioned anywhere on a Web page using absolute or relative positioning. Z ordering allows for objects to be overlaid so they can be layered within a Web page. Objects can also be moved to create animations within a Web page, with text and graphics flying all over the page.
Data Binding – Can be used at run time to create simple client-side databases of information that can be retrieved locally from the client instead of the server.
Dynamic Content – Content can be dynamically changed on the fly. Objects can be made invisible, or text and graphics can be changed based on client-side scripts.

Dynamic Fonts

Netscape includes dynamic fonts as part of dynamic HTML. This feature of Netscape's Navigator* browser in its Communicator* suite lets Web page designers include font files containing specific font styles, sizes, and colors as part of a Web page and to have the fonts downloaded with the page. The font choice is no longer dependent on what the browser provides.

*Appendix D – WebCD Publisher**

*WebCD Publisher**

Developers wanting to convert an existing Web site to a hybrid application should consider using a tool like WebCD Publisher*. WebCD Publisher* is used to create a deliverable CD-ROM with Web pages or assets stored locally mixed with assets residing on the Internet. WebCD Publisher* allows developers to manage the process of creating a CD with better-quality video, audio or graphics delivered with the application.
WebCD Publisher* provides an easy-to-use tool that is as simple as surfing your site and then redirecting URL links to assets residing on the CD-ROM. The tool also provides a built-in search engine and bookmarking feature for making it easier to navigate through your site. The WebCD* architecture includes an ISAPI interface, which makes customization of your CD application easier. A kiosk setting allows for a branded look and feel, with the browser's toolbars and menus eliminated.
The tool provides a comprehensive set of features to create and manage a Hybrid CD application. The game channel application was created using the WebCD Publisher, and the information in chapter 5 covers step-by-step creation of a WebCD Publisher-based application. A potential disadvantage of WebCD Publisher is that the current version cannot be used to integrate Internet and CD-ROM data with data delivered by push technologies
WebCD* Publisher provides a hybrid engine for accessing content on the Web, CD-ROM or hard disk. The engine intercepts browser requests to the server and redirects the content, depending on which version—CD-ROM, Web or hard disk—is newest. The engine uses the HTTP proxy interface to process and redirect references.

Smart Caching

The WebCD* Publisher makes use of a content filter to determine where assets are retrieved. The WebCD* Publisher viewer uses the last modified date stored for assets. If one does not exist, it will generate one with the date of the asset stored on the CD-ROM. The date of the asset is compared between the server and the CD-ROM, and only the newest content is retrieved. With this feature, applications can mix and match assets from the CD-ROM with newer assets on the Web.

Source Blending

The WebCD* Publisher provides for source blending by using the expiration date on HTTP documents and assets. Large assets such as graphics and Indeo Video clips are stored locally on the CD-ROM. This makes for a better user experience, since better graphics and full-screen video can be stored on the CD-ROM. Assets can still be stored on the Internet server and provide real-time information. An example used in the game channel is advertising banners. These assets are not stored locally on the CD-ROM but are

retrieved from the Web server. The latest advertising for the newest games will always be displayed and can provide links to Web sites where the newer games reside.

Push Technology

Currently, WebCD* Publisher does not incorporate push technology for updating assets to the user locally. Push technology support is an area for future enhancement.

Internet Connection Manager

The WebCD* Publisher viewer offers two modes of operation: Fresher and Faster. This allows the user to access content off-line or online. Web site content can be viewed entirely off-line. Improved performance from off-line browsing is one of the biggest advantages to using channels.

ISAPI Interface

The WebCD* Publisher hybrid engine includes an ISAPI interface with which the developer can write extensions. URL references can be intercepted and processed by the application. One use is to track and store user selections in database user preferences. These preferences could be reported back to the server to provide content or information for advertising based on the user's interests. Local content and information could be retrieved to better target the user.

In the case of the game channel, game or advertising information could be based on the user preference for action games, for instance. With the large number of games available on the market, it is important that the user be able to be presented with information that targets his or her interests.

As an example of the ISAPI interface, the WebCD* Publisher viewer comes with a search engine that was implemented as an ISAPI extension. The search engine allows the user to quickly locate documents stored on the CD-ROM. Being able to search content is especially important with channels that contain a lot of information.

Bookmarks

The WebCD* Publisher includes a bookmark feature that allows the developer to generate key links to the CD-ROM or link to the Web. The Game channel contains bookmarks to each major game as well as other important links to information sites. Links to information on WebCD* Publisher are in bookmarks.

Bookmarks are easy to add to your Web CD-ROM. You just mark the URLs as bookmarks as you browse a Web site.

The BackWeb* solution consists of a BackWeb server running on a Windows NT* Server and the BackWeb* client software that manages the delivery of BackWeb InfoPaks and subscribes to BackWeb channels*. This is a very flexible push solution for updating software residing on the client.

Appendix E –Push Technology

*BackWeb Server**

The BackWeb* server provides for the management and delivery of updates using the BackWeb* server console. The BackWeb server console manages the delivery of InfoPaks from the server to the client. InfoPaks consist of screen savers, animated graphics, Java applets, Shockwave* animations and audio messages. The BackWeb server console allows the system administrator to create channels and then define InfoPaks for delivery. The console also sets properties such as version number, time of delivery, number of times to send an InfoPak, and so on. The BackWeb* server schedules and manages InfoPaks for delivery to the end-user client.

InfoPaks can also be targeted based on user preferences. The BackWeb* server can deliver InfoPaks based on criteria set up at user registration or stored in a database. This allows for custom delivery of information depending on the user's interests. The BackWeb* server will deliver content based on channel criteria.

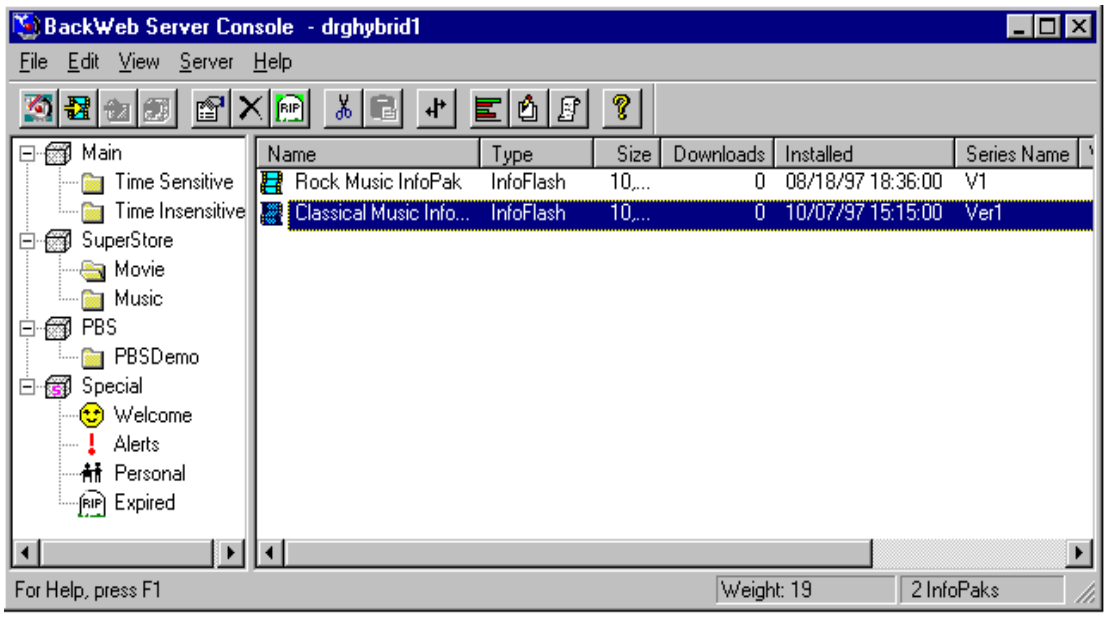

Figure E-1. BackWeb Server Console

InfoPaks

InfoPaks come in four forms: wallpapers, infoflashes, audio and screen savers. An InfoPak can contain any number of files, including executables. The size of the InfoPak can be quite large, although the BackWeb client allows the cache size to be limited by the user. InfoPaks are delivered when the client subscribes to the channel.

Bali Editor*

InfoPaks can be created using the Bali editor, which provides a basic scripting language for handling the notification and delivery of InfoPaks. The editor can design scripts that run Java applets* or call up Web sites on the Internet.

Sample Advertising Announcement Script

The following sample script runs a simple text animation announcing a special deal on software; when clicked it accesses the Web site for online ordering. This notification is a simple example of advertising in which the BackWeb InfoFlash* mechanism announces a special on a product.

```
Start {
        set frameFile=default
        set reference=midBottom
        prepareText text=" HOT DEAL – Starship Madness AT 29.95"
         sequence=0
         font="Courier New" size=20 bold
         fgColor=green transparent
        accessWeb URL="http://134.134.148.117/softsuper/store.htm
        onClick {

                accessWeb URL="http://134.134.148.117/softsuper/store.htm
                # TODO: Add more rectangles
        }

        animate from (-300,1000) to (500,1000) in 3 seconds
        pause for 10 seconds
        animate to (1300, 1000) in 3 seconds
}

Open {
        }
```

Figure E-2 Sample Bali Script to add advertising banner

BackWeb Client

Users can subscribe to BackWeb channels using the BackWeb client*. They can set up the size of the cache for downloading InfoPaks and define which types of InfoPaks to load as well as in what order. InfoPaks can also be launched automatically or selectively by the user. InfoPaks delivered to the client can be automatically deleted when set to expire by the server. InfoPaks can be delivered to the user without affecting Internet browsing.

Polite Agent

The BackWeb client polls the BackWeb server for new InfoPaks. The BackWeb server downloads InfoPaks in the background when the user is connected to the Internet. The BackWeb server downloads InfoPaks only when the client's Internet connection is idle. Downloads are transparent as new InfoPaks become available.

Marimba*

Marimba provides the Castanet tuner* for delivering push data to users. Castanet* channels can consist of software updates, HTML pages or Java applications. The basic Marimba solution consists of a transmitter, a tuner and a publisher. Marimba was primarily designed to deliver networked applications to the client. Castanet can be used as the primary push mechanism for delivering updates to Netcaster* channels. Marimba can deliver incremental updates to the software application, efficiently downloading only the changes that have been made in the application.

Marimba Castanet Transmitter*

The Marimba Castanet* transmitter is relatively easy to set up. It requires setting up the simple directory where the channels are stored, the IP host address and port number, and a few other parameters. Once the channel is launched, the publisher is used to create content that can be fed into the transmitter. The transmitter is efficient because only modified files are sent across the network.

Marimba Bongo*

Marimba provides Bongo* for developing applications for the Castanet channels. Bongo creates presentations that are essentially the integration of GUI components and scripts into an application. The GUI components or widgets can be pulled into an application. Bongo provides a variety of widgets, such as menus, text, buttons, graphics, sounds and so on. These widgets can become a full-blown Java applications that can then be downloaded incrementally and broadcast using the Castanet transmitter.

Marimba Publisher*

The publisher sets the actual channel data that needs to be published through the transmitter. Publishing data can be as simple as specifying the directory where the data resides and then naming the channel. The publisher interfaces with the transmitter to send the data according to rules and parameters set in the publisher. The Marimba publisher *defines the HTML pages, Java applications and standard files to be transmitted over a Castanet* transmitter.

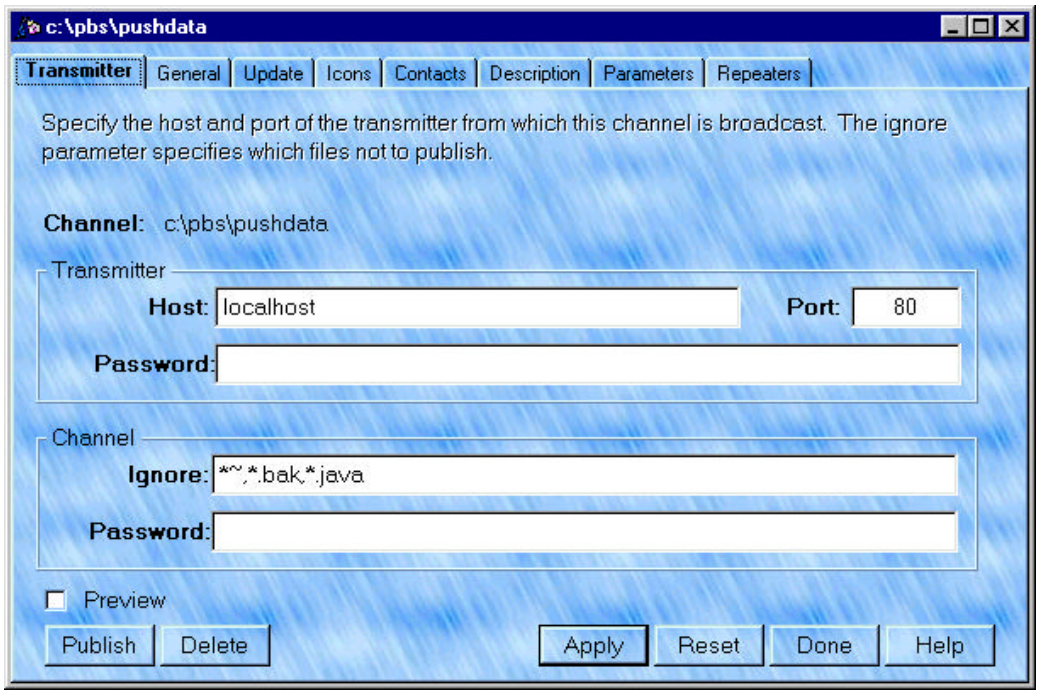

Figure E-3 Marimba Publisher*

Marimba Receiver*

The client software allows the user to subscribe to channels and manage incoming data. Once they have subscribed, users can receive applications that are downloaded and can be launched automatically. Updates to the client can be determined by the author based on the content. Updates to the channels can come automatically or selectively by the user.

Summary

We have covered a variety of methods for managing multimedia assets. The method chosen depends on the application. Push technology is an important component of channels. Whether using BackWeb, Marimba, Internet Explorer, or Netcaster, push technology can enhance channels by delivering software updates and content to users across the Internet*. Users can subscribe to channels and set preferences for receiving content targeted to their interests. Users can then be sent notifications of live events and receive targeted advertising as well as the latest and greatest content. Data management of assets ultimately improves the entertainment value of a channel by providing the best possible video, audio or 3D graphics integrated transparently into an easy-to-use interface. Providing better quality and scaleable assets is important for channel developers, to ensure the success of channels.

Appendix F – The Internet Connection

A digital content channel requires a network infrastructure capable of delivering a variety of content to the end user. A typical channel requires servers for delivering Web pages, pushing content, streaming real-time video or audio, or providing online chat. The networks for channels also need to be scaleable to be able to service a large number of users on the Internet. This requires that the server technology being used is highly scaleable. The network must also have sufficient bandwidth to service the traffic that will be generated by a successful channel.

The Internet as seen from the home user is typically a modem connection into an Internet Service Provider (ISP). The ISP leases a T1 or larger line from a major carrier such as MCI* that actually provides an interface to the Internet backbone. The Internet consists of routers that store in its tables the actual route of the IP addresses going across it. These routers can handle a large amount of traffic but are the primary cause of the latency that a user encounters. During peak hours, this latency can exceed one second or more. And in some cases, latency increases when internet requests time out or packets are dropped. When the user dials into an ISP, he or she may expect a full 28.8-Kbps connection. This speed is, of course, dependent on the capabilities of the ISP and the quality of the line connection. The actual speed of the connection may be lower in some cases. This connection is, of course, shared with a large number of users who connect through the T1 line. And then the backbone is shared among an even larger number of users.

The destinations for IP requests are servers that respond to the request for a particular packet. The large numbers of requests that come in can overwhelm any server. Any popular site can experience server busy messages, depending on the load currently displayed to the user. For channels, it is important that server technologies being used are scaleable. For channel developers, it is important to understand the limitations of the Internet to determine what rich multimedia data types can be realistically delivered over the Internet.

The consumer has a variety of options (in some cases) for connecting to the Internet. These options include 14.4- to 56.6-Kbps modems. New technologies are emerging that provide higher bandwidth for the end user. These technologies include cable modems, satellite and ADSL**.** All promise to deliver high-bandwidth pipes to the consumer that should ultimately benefit channels.

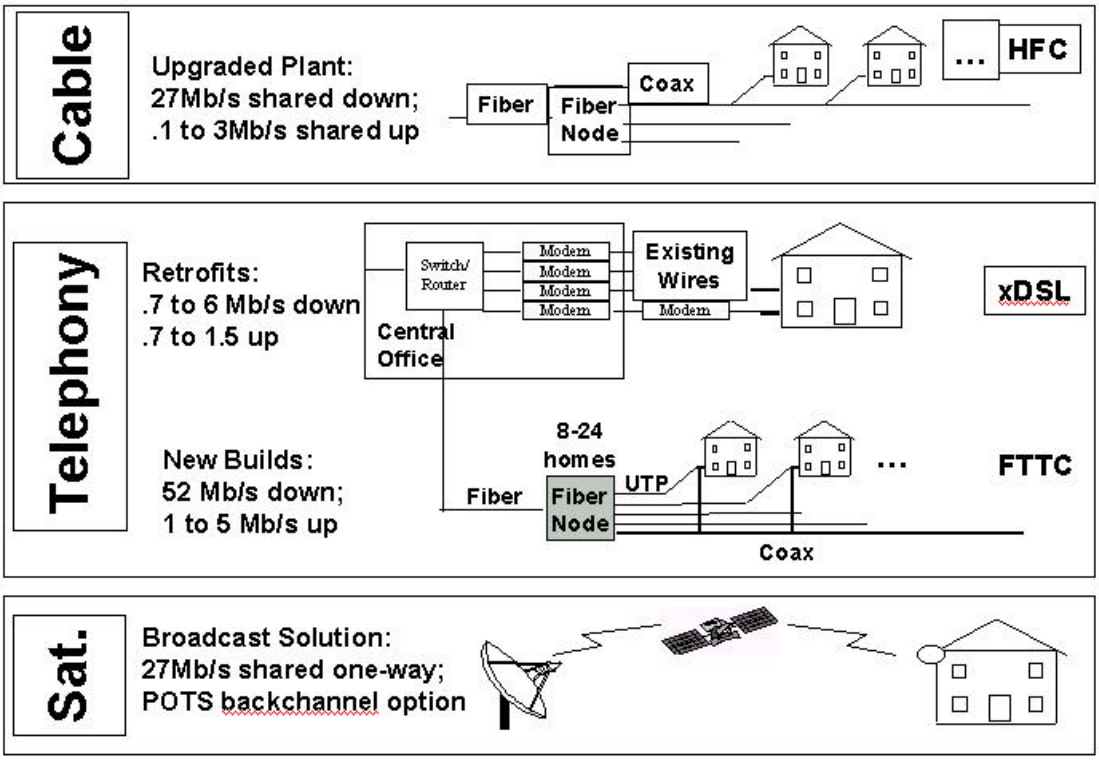

Figure F-1: Future Internet connections to the home

Modems

A majority of the modems currently connected to the Internet run at 28.8 Kbps. Newer 56-Kbps modems are widely available but require support at the ISP level. This effectively doubles the potential bandwidth for users dialing into the Internet over standard phone lines. The actual modem bandwidth is highly dependent on the quality of the phone connection and the availability of the higher speed modem at the ISP. Modem speeds can drop down to 19.2 Kbps or lower. The effective bandwidth can be somewhat lower, depending on network traffic at the time. Internet traffic at peak times during the day tends to occur in bursts and can further limit effective bandwidth.

Given a 28.8-Kbps modem connection, the user can receive very limited video and good-quality audio in real time over the Internet. Most Internet Web sites are limited to 2D graphics with text. Video is restricted to 160x120 with low frame rates. Audio is somewhat better, with quality that approaches what one gets with FM radio. This type of audio is good for live concerts and interviews with band members. This limitation can be overcome with push servers to increase the size of assets that are delivered, since this is done in the background. However, applications can benefit greatly with better quality audio, video, 3D and graphics delivered on storage media such as CD-ROM and DVD-ROM. Some emerging technologies may increase the bandwidth available on the Internet. ADSL is one of these higher bandwidth solutions becoming available for the home.

ADSL

ADSL is an emerging technology that can provide anywhere from 1.5–9 Mbps downstream and 16–640 Kbps upstream when connected to the Internet. The connection speed of ADSL is dependent on the distance from the Telco. This limits the availability of ADSL to the home and the maximum speed that can be attained with ADSL. At 18,000 feet from the Telco, the maximum bandwidth is limited to 1.544 Mbps. The adoption of ADSL is currently not widespread, but it is a technology that is capable of providing fast access to the home, since it connects over standard twisted-pair telephone lines.

ADSL provides much larger bandwidth to the end user and can potentially deliver real-time MPEG1 and MPEG2 streams and virtually any high-quality audio compression schemes, such as AC3. This is as long as the high-bandwidth streams can be delivered across the Internet backbone and the infrastructure provided by the ISP. ISPs typically have T1 or T3 access to the Internet backbone shared by multiple users. It is highly unlikely that the ISP would dedicate the equivalent of one T1 line to deliver a stream of video to one end user. The ISP can provide multicast broadcasts, but the ISP is limited by the bandwidth available to the Internet. Private networks are being put in place to deliver multicast streams on the Internet, but their deployment is currently limited.

ADSL shows some promise for delivering high-bandwidth connections to the home. Of course, improvements in the capacity of the backbone, quality of service and ADSL equipment at the telephone company will make the availability of ADSL limited to certain areas in the short term. Cable modems are another technology that unlike ADSL already has connections to the home.

Cable Modems

Cable modems have been in existence for several years, and available cable modems and head-end solutions are starting to mature. Cable modems use what is known as Hybrid Fiber Coax (HFC) to deliver high-bandwidth data. The cable modem can deliver up to 30 Mbps of downstream data to the end-user PC. Users share from 0.1–3 Mbps on the upstream link. Alternatively, the end user can utilize an existing modem connection to the Internet as an upstream link.

Cable modems require a head-end solution. The head end equipment resides at the cable company and provides the router, modulator, network management software and an upstream server switch. This head-end system is shared by thousands of users. Cable modems provide a large bandwidth to the home and allow for the potential delivery of good-quality MPEG2 video as well as a large amount of data. The main advantage of cable is the availability of connections to a large number of existing homes. The large bandwidth makes it a potential solution for delivering digital channels to the home.

The large number of cable operators across the country makes it difficult to deploy to all parts of the country simultaneously. The deployment of cable modems has been slowed because of the investment required by cable companies.

Satellite Broadcasts

Satellite broadcasting is an emerging technology that holds great promise for the delivery of digital data. Its main advantage is that it can be delivered from one satellite system and

reach a large number of users simultaneously. Investment in head-end equipment is cheaper, since a stream is broadcast from one satellite to every subscriber.
Satellite provides up to 27 Mbps on the downlink, with uplink being provided by the modem. A large amount of data can be delivered to the PC continuously. This is 1,000 times more bandwidth that can be delivered by a standard 28.8-Kbps modem. It is an ideal medium for delivering broadcast data channels that can be received by millions of users.
One example of a satellite connection is DirectPC*. It delivers 400 Kbps one way to the home. Upstream is done using a standard telephone modem with a speed of 33.6 Kbps. This provides Internet access with a higher bandwidth available on the downstream. DirectPC* is currently available to any home that can install a satellite and with a direct line of sight (no trees) to the satellite.
The advantage of satellite is that it can be sent data that would not have to travel over the Internet to arrive. It could be sent directly to the satellite head end and broadcast immediately to users. Satellite deployment is limited today but is growing rapidly.

Summary

Surfing the Internet at speeds of 1–6 Mbps seems unthinkable to most users. However, the speed of connection to the Internet is not always the determining factor of Internet delivery speed. The backbone limits the amount of data that can be effectively delivered online at any given time. The large amount of traffic is shared with every user on the Internet.
In addition, popular sites can be bogged downed with the common server-busy message. The main point is that connecting to the Internet at 1 Mbps or higher will not guarantee that type of service. Traveling across the backbone limits the effective bandwidth at which users can surf the Internet. Some T1 connections have the throughput of a 28.8-Kbps modem, depending on Internet traffic.
The bandwidth available for delivery of rich multimedia across the Internet will be limited in the short term. To deliver rich multimedia and exciting channels to the user requires the use of CD-ROM, DVD-ROM or push technology. The future holds great promise for delivering rich multimedia channels with the wider availability of ADSL, cable modems and satellite. The key for digital content channel developers is to provide scaleable solutions that take advantage of higher bandwidths. Appendix B covers live streaming of audio and video to enhance a channel, despite current limitations in bandwidth. Several scaleable solutions are available that take advantage of increasing bandwidth and richer data types.

Appendix G – Server Hardware and Technology

Server Hardware

Server hardware requires fast performance and large storage in order to deliver content to a large number of users. Windows NT* Server-based servers built around Pentium® II processors provide a high-performance server solution at a reasonable cost. The server should contain as much memory as possible to handle the large number of transactions that are possible. A 100-Mbit high-speed network ensures that the network can handle large amounts of traffic, and 100-Mbit Ethernet cards provide the much-needed increased bandwidth and the performance necessary for servers to run effectively. Servers also require a large amount of storage, and Fast SCSI II hard drives provide the speed necessary to handle a large number of storage accesses. Pentium® II Processor based servers provide the needed performance to deliver the scaleable solutions required by channels.

Pentium® II processor
64 megabytes of memory
Windows NT* 4.0
100-Mbit network card
Fast/Wide SCSI II hard drive

Internet Connection Required for a Channel Server

Each channel must connect to the Internet using a router and a T1 or T3 line out to the Internet. Routers essentially route packets coming from the Internet to their appropriate destinations. Routers need to be able to handle large amounts of traffic in order to support a channel.

Router – A router is required to handle the incoming packets and deliver them to their destination.
T1 Line – Provides 1.54 Mbps of bandwidth available to the user. This typically is the connection out to the Internet.

Web Servers

A Web server is required to deliver HTML pages to the user. In addition, the server needs to support a variety of scripting languages, such as CGI, Java and ISAPI. The server must also be capable of delivering security to control access to documents, support SSL and password authentication, and be able to handle security certificates. Servers must also be able to log access and page hits, interface with online databases, and provide general configuration and management. The Web server is the heart of a channel and provides real-time access to thousands and, in some cases, millions of users. To handle the large

number of users, Web servers need to be scaleable. The server selected for our three channel applications was Microsoft's Internet Information Server.

Chat Servers

Another compelling component for real-time interaction is chat. Chat is typically used as a forum for people to discuss specific items or agendas. Chat servers have prompted the growth of many online communities that have come together to share thoughts and feelings. The types of chat servers include voice, text and 3D worlds.

A simple text chat solution consists of a chat server and a chat client. The server is typically written using TCP; however, it is possible to implement a UDP solution. The server is responsible for listening for requests for connections, accepting connections and forwarding messages to the intended recipients.

The chat client is a simple interface that requests a connection from the chat server. Once a connection has been granted, all messages are forwarded and received from it.

E-Commerce and Advertising Servers

From a content provider's viewpoint, the most important aspect of a channel is its ability to generate revenue. Channels can create revenue by providing online catalogs of products that can be purchased electronically or product advertising. After visiting a channel and viewing the latest movie or music album, users need to be able to purchase merchandise online.

A variety of electronic commerce servers can be used to create online stores. The online store needs to provide catalogs for viewing and searching for various products. A comprehensive electronic commerce solution provides the order tracking, shipping, inventory, pricing and interfaces required by online electronic commerce.

In addition, the commerce server needs to be able to process the user's credit card and address information. Solutions such as Microsoft Wallet* or CyberCash* can be used to generate cash transactions on the Internet.

Advertising is also a key component to the success of a channel. To help promote the sale of merchandise, advertising banners can be included. Ad banners can be delivered to the users based on their preferences. Targeted advertising has the benefit of delivering only advertising about products that the user is interested in buying. This form of targeted advertising is ideal for companies wanting to make product sales.

Both advertising and electronic commerce are dependent on databases that contain most of the product information and/or user preferences.

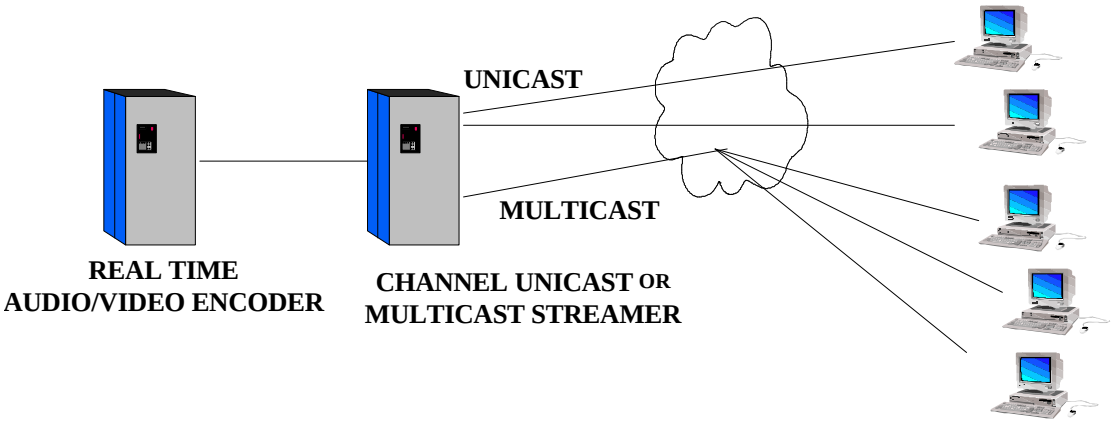

Figure 5-3. Audio and Video Streaming

Streaming audio and/or video across the Internet requires a real-time audio or video encoder. A video and sound capture card is required as well as a server for encoding the stream in real time. The output of the encoder is typically fed into a channel manager. Its role it to allow multiple users to connect to the server and retrieve the broadcast stream using unicast or multicast. Multicast has the benefit of requiring a single stream, compared to one stream for each connected user.

IP Multicast

To deliver one stream to many users, a network must be capable of handling multicast IP. This requires that routers be programmed to support multicast packets. Packets being delivered across the Internet will travel through several routers from server to client. Each router on the network must be programmed to support multicast packets. Most routers on the Internet are not programmed to support multicast. To deliver a multicast stream, a multicast-capable network must be available to handle multicast packets.
Packets moving across the Internet also require some type of quality of service. Since the bandwidth on the network is shared, video or audio streams are not guaranteed the amount of bandwidth they require. RSVP can be used to reserve bandwidth on the Internet. This would work if all routers supported RSVP and were configured to do just that. This is not the case. However, for private networks, the routers can be programmed and quality of service could be guaranteed. For today's Internet, channel developers will require the use of unicast for delivery of live video or audio across the Internet.

Unicast

Most broadcasts across the Internet are generated as point-to-point unicast transmissions from server to client. Most streaming audio and video server solutions support unicasts.

Depending on the channel, the number of unicast streams supported will require at least a T1 or T3 line to handle the multiple streams. For example, a T1 line at 1.5 Mbits/sec could handle about 50, 28.8-Kbps streams on the Internet.

Streaming Audio

A variety of streaming audio codecs provide audio across the Internet. Streaming audio can be used in channels to provide live concerts, artist interviews, presentations, live radio broadcasts and music clips. Creating a streaming audio codec requires an audio server and an audio source. A Windows NT* server can be used to configure and manage the audio to be streamed on the Internet. Real Audio* and Xing* provide scaleable audio servers for delivering live audio streams across a network. Several other streaming solutions are also available. The reference section at the end of this white paper lists some possible solutions.

Database Servers

Databases are needed to support the electronic commerce and advertising required by channels. Databases are used in electronic commerce to keep track of orders, customer purchases, catalogs of merchandise, pricing information, inventory and so on. Advertising databases can be used to deliver advertising banners that are targeted to user preferences. Pushed content such as movies and music can also be delivered based on user preferences. If the user is interested in action movies, advertising or movies presented in the channel should be tailored to action movies. Databases can also be set up to deliver content based on user demographics. Viewers can be presented with local events, such as movies opening up and concerts occurring locally, based on ZIP codes.
Databases are also used to track user selections and preferences. The collection of information about a user's choices while in a channel benefits the user as well as the channel provider. The channel provider gets a database tracking the buying and selection habits of subscribers, and the user benefits from getting content tailored to his or her interests. Users participate in channels according to interest, and they can be connected to other users who share those interests.

Summary

Channels require a variety of network servers to provide real-time content and updates to users. The integration of different technologies into a rich multimedia application is a big challenge for developers. Appendices F and G describes two potential channels: a music application and a theater application. The appendix describes the servers and their application in channels.

Appendix H - Channel Server Components

*Microsoft Information Server**

Many servers on the market are available for use as Web servers. In developing our digital channel applications, Microsoft's Internet Information Server* (IIS) provided useful features that could be used in channel applications. Active Server Pages (ASPs) were used to deliver compelling content.

A Web server is required to deliver HTML pages over the Internet. In addition, the server should support a variety of scripting languages, such as CGI, Java* and ISAPI. The server must also be capable of delivering security to control access to documents, support SSL, provide password authentication and be able to handle certificates. Servers must also be able to log access and page hits, interface with databases and provide general configuration and management. The Web server is the heart of a channel that provides real-time access to thousands, and in some cases, millions of users. It must be scaleable to handle large volumes of traffic. The server chosen for creating our three applications was Microsoft's IIS.

Active Server Pages

Microsoft's IIS* provides a full-featured solution for delivering HTML pages on the Web. Also included is support for ASPs, which allow for the server-side execution and delivery of dynamic content. ASPs were used to develop the Microsoft Merchant Server for our electronic commerce solution. ASP scripts were written to resolve asset resolution and provide an e-commerce solution incorporating richer media data types. Rather than using standard graphics, the catalog could mix video assets with online commerce. Doing so allowed channels to deliver videos on CD-ROM that could be previewed and then immediately purchased on the Internet.

Microsoft Merchant Server*

An electronic commerce solution consists of providing an online store for consumers to connect over the Internet. An electronic commerce Web site must provide the ability for the shopper to look up product information. In addition, the Web site must be able to process incoming orders, store the latest price information and add any applicable tax. The e-commerce solution must also be able to track customer orders and product inventory, and process credit card and shipping address information. One alternative that provides the features necessary to create an e-commerce solution is Microsoft Merchant*.

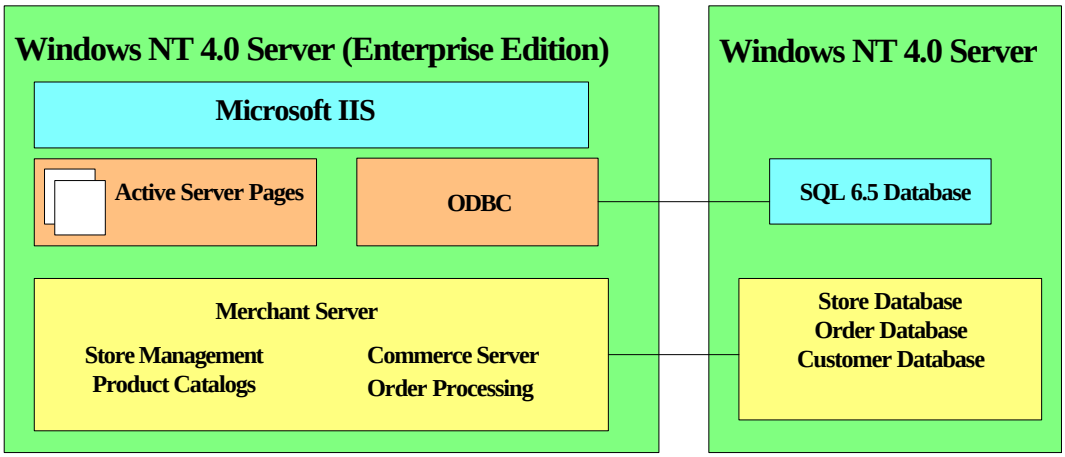

Figure C-1. Microsoft Merchant Server Architecture

Merchant Server Architecture

The Microsoft Merchant Server architecture is built on a Windows NT* Server. The solution consists of a IIS Web server for processing Web pages and ASPs. ASPs allows server-side scripts to be written that provide general, customizable and dynamic HTML pages that can be delivered to the client. The Merchant Server* makes use of an SQL database server to store products, customer and order information, as well as process credit card and address using Microsoft Wallet* on the client side.

Microsoft Merchant* Server Setup

Setting up the Microsoft Merchant Server requires the integration of three major components: Internet Information Server, SQL Server, and the store setup itself. Internet Information Server needs to be set up to handle processing of HTML Web pages and support for ASPs. The SQL Server database software needs to be set up prior to loading up the Microsoft Merchant* Server. Several Merchant Server databases need to be set up using SQL Server Enterprise Manager*. The procedures entail setting up the data devices and required database, and the documentation included with the Microsoft Merchant* Server takes users through setup step by step.

After setting up the SQL Server database, the next step is to run through the Merchant Server setup. This involves setting up the interface to the SQL Server database and then creating the directory structure for the actual store. Internet Information Server must also be properly set up to handle secure access to the online store.

After setup is complete, all the ASPs, which make up the store, are loaded. These files and directories can be modified to provide your own look and feel. Graphic screen elements need to be changed and product information replaced with information about your own products. Customization of ASPs can be done to meet your application needs.

Merchant Server includes several tools to help create and manage online stores. A store foundation wizard and store builder wizard can be used to generate a store. Or you can use one of the four templates provided by Merchant Server. We used the bookstore template to help create a software superstore.

An online, Web-based interface is used to access the product database and input new products. When product descriptions, prices and other information are entered, you have the template for an online store.

Online Multimedia Catalog

The stores included with the Microsoft Merchant server provide two-dimensional graphics for the products. ASPs provide the flexibility to add short video descriptions of the product. Intel Indeo Video 5.0 is an excellent choice for providing either high-quality video clips that could be distributed using a CD-ROM or storing video clips online. Video clips could also be pushed to play off the users' hard drives.

Microsoft Address Selector and Payment Selector ActiveX Controls

Microsoft Wallet consists of two ActiveX controls for providing credit card information from the client. The store wizard creates four active server pages for processing of this information from the client. These files can easily be customized to process the customer information for an incoming order.

Microsoft Select Language Query Server

The SQL server is the database component for the Microsoft Merchant server. The databases provided are for tracking orders, inventory, store catalogs, pricing information and so on. The Microsoft Merchant Server uses ASPs to query the SQL databases for information to be displayed back to the user.
Databases are useful for storing archived information. In the music application, Microsoft Access was used for the music database, which contained information about albums, artists and specific tracks.

Summary

Channels benefit greatly from the use of electronic commerce to provide revenue streams from selling online products. The distribution and purchase of software, videos and music will become more prevalent as bandwidth increases, but until then, users can still purchase items online. Channel developers can choose from a variety of electronic commerce solutions in creating an electronic store, and CyberCash* and/or Microsoft Wallet* can then be used to facilitate electronic transactions across the Internet.

**Other brands and names are the property of their respective owners.*