# Video servers

A. L. Narasimha Reddy
Texas A & M University
214 Zachry
College Station, TX 77843-3128
reddy@ee.tamu.edu

Roger Haskin
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120.
haskin@almaden.ibm.com

**Abstract**

The recent dramatic advances in communications technology has led to a demand to support multimedia data (digitized video and audio) across data networks. In part this demand comes from the personal computer marketplace, where the use of video and audio in PC applications and games is now pervasive, and in part the demand comes from the entertainment marketplace, where there is a move to use digital video to supplant the traditional analog methods of content creation and broadcast. This intersection of technologies and markets has led to the development of videoservers.

# 1 Introduction

Several telephone companies and cable companies are planning to install video servers that would serve video streams to customers over telephone lines or cable lines. These projects aim to store movies in a compressed digital format and route the compressed movie to the customer (through cable, telephone lines or local area networks ) where it can be uncompressed and displayed.

These projects aim to compete with the local video rental stores with better service; offering the ability to watch any movie at any time, avoiding the situation of all the copies of the desired movie rented out already, offering a wider selection of movies, offering other services such as tele-shopping and delivery of other video content such as interactive video games.

A number of business models have been proposed for such video servers. The business models include complete interactive video games, near-video-on-demand in hotels among others. The systems' requirements vary considerably depending on the targeted markets. Near-video-on-demand envisions supplying popular/recent movies at regular intervals, say every 5 minutes, to requesting customers over some delivery medium. This form of service is expected to be popular in hotels, replacing the existing situation of making movies available at only few times during the day. Few tens of movies are expected to be available through such a server. Video-on-demand (VOD) envisions supplying a larger number of movies to customers whenever the customer requests the movie. These video servers provide access to a larger number of movies to possibly larger number of customers to amortize the costs of a large system. Some of these servers plan to provide the ability to provide VCR-like capabilities to pause/resume, fast-forward/reverse a movie stream. Such servers are said to provide interactive TV (ITV) service. Interactive video servers could have wide applications. Some of the planned applications of interactive video servers include training, education, travel planning etc. These interactive video servers play a predetermined set of video sequences based on the user's input. Interactive video games require a higher level of interaction with the video server since the playout of the video sequence varies more rapidly based on the user's input. Video teleconferencing requires two-way interactive response between the users. Which of these several business models will survive remains to be seen.

Digital Video Broadcast (DVB) systems replace analog video channels with compressed digital video. DVB allows more channels to occupy the same frequency spectrum, and allows expensive and failure-prone electromechanical devices (e.g. video tape players) to be replaced with more reliable digital storage. In a DVB system, the video storage and transmission are identical to those in a VOD or ITV system. In DVB, however, instead of the video being under control of a user, a piece of automation equipment controls the sequencing of video playback. Here, "sequencing" means creating a smooth, continuous video stream out of multiple independent pieces of stored video, e.g. a series of commercials followed by a segment of a TV program, a station break, more commercials, etc. A DVB system is actually in this respect more comparable to an ITV system than to VOD, since the system is required to respond in timely fashion to commands from the automation equipment.

A number of pilot projects have been underway to study the feasibility of these approaches. Several major companies in computer systems business (IBM, SGI, HP), computer software (Oracle, Microsoft, IBM), telephone carriers (AT&T, US West, Bell Atlantic, Ameritech), cable companies (TCI, Time Warner, Cox cable) and equipment manufacturers (Scientific Atlanta, General Instrument) have been actively participating in these trials. The technical challenges in making these projects economic/business successes are daunting. In this chapter, we highlight some of the technical problems/approaches in building such videoservers, with an emphasis on communication aspects of the problem.

Currently, with MPEG-1 compression at 1.5Mb/s, a movie of roughly 90 minute duration takes about 1 GB of storage space. Storing such large data sets in memory is very expensive except for the most frequently requested movies. To limit the costs of storage, movies are typically stored on disks. It is possible to store infrequently accessed movies on tapes or other tertiary media. However, it has been shown that these media are not as cost-effective as magnetic disk drives even for infrequently accessed movies [1].

Providing a wide selection of movies requires that a large number of movies be available in digital form. A video server storing about 1000 movies (a typical video rental store carries more) would then have to spend about $250, 000 just for storing the movies on disk at the current cost of $0.25/MB. If higher quality picture is desired and MPEG-2 compression is used at 6.0 Mb/s, then the cost of disk space goes up to $1,000,000. To this, the cost of computer systems required for storing this data needs to be adeed. Other non-recurring costs include costs for enhancing the delivery medium, providing a set-top-box for controlling the video delivery. This requirement of large amounts of investment implies that the service providers need to centralize the resources and provide service to a large number of customers to amortize costs. Hence, the requirement to build large video servers that would store a large number of movies in a single system and be able to service a large number of customers. However, the ever decreasing costs of magnetic storage (at the rate of 50% every 18 months), and computing power (at the rate of 54% every year) may make these cost considerations less of a problem if the technical challenges can be overcome.

## 2 Data Server

A large video server highlights most of the technical issues/challenges in building a video server. A large video server may be organized as shown in Fig. 1. A number of nodes act as *storage nodes*. Storage nodes are responsible for storing video data either in memory, disk, tape or some other medium and delivering the required I/O bandwidth to this data. The system also has *network nodes*. These network nodes are responsible for requesting appropriate data blocks from storage nodes and routing them to the customers. Both these functions can reside on the same multiprocessor node, i.e., a node can be a storage node, or a network node or both at the same time. Each request stream would originate at one of the several network nodes in the system and this network node would be responsible for obtaining the required data for this stream from the various storage nodes in the system. The design of network nodes will change based on the delivery medium. The logical separation (whether or not they are physically separated) of nodes into network nodes and storage nodes makes it easier to adapt the videoserver organization to different delivery mediums.

The delivered data stream is fed to a set-top-box(STB) at the customer's site. The STB decompresses the digitally enoded data into a suitable input form for the TV monitor. The STB may provide additional features such as the ability to pause/resume, fast-forward/reverse an incoming video stream. The design of the STB is a hotly discussed topic with proposals ranging from a dumb decoder to intelligent controller with most of the functions of a personal computer. The design of the STB depends on the functions provided, compression standard, bandwidth of the compressed stream, input standard of the TV monitor among other things. To enable future adaptation to evolving compression technology, some projects envision doing decompression completely in software by a processor in the set-top-box.

To obtain high I/O bandwidth, data has to be striped (distributed) across a number of storage nodes. If a movie is completely stored on a single disk, the number of streams requesting that movie will be limited by the disk bandwidth. To enable serving a larger number of streams of a single movie, each movie has to be striped across a number of nodes. As we increase the number of nodes for striping, we increase the bandwidth for a single movie. If all the movies are striped across all the nodes, we also improve the load balance across the system since every node in the system has to participate in providing access to each movie. A number of related issues such as fault-tolerance, incremental growth of the system are closely linked to striping characteristics and we will discuss these issues later in the chapter. The unit of striping across the storage nodes is called a block. In earlier studies on disk scheduling [2], 64-256 Kbytes is found to be a suitable disk block size for delivering high real-time bandwidth from the disk subsystem.

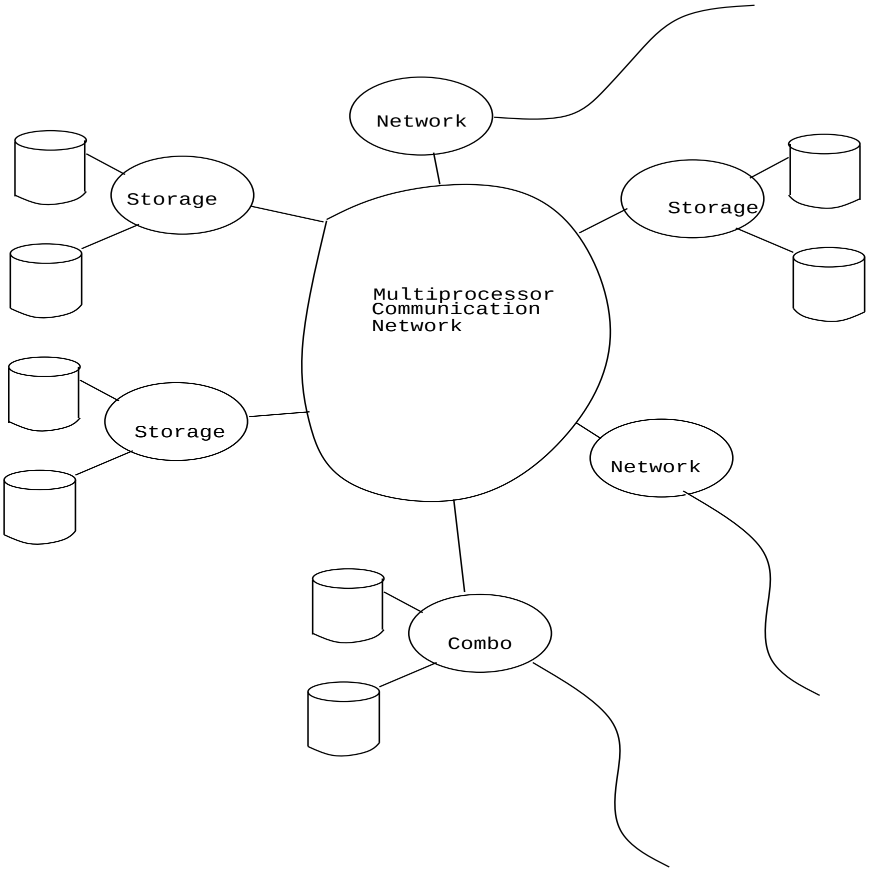

Fig. 1. System model of a multiprocessor video server.

As a result of striping, a network node that is responsible for delivering a movie stream to the user may have to communicate with all the storage nodes in the system during the playback of that movie. This results in a point to point communication from all the storage nodes to the network node (possibly multiple times depending on the striping block size, the number of nodes in the system and the length of the movie) during the playback of the movie. Each network node will be responsible for a number of movie streams. Hence the resulting communication pattern is random point-to-point communication among the nodes of the system. It is possible to achieve some locality by striping the movies among a small set of nodes and restricting that the network nodes for a movie be among this smaller set of storage nodes.

The service for a video stream can be broken up into three components: (1) reading the requested block from the disk to a buffer in the storage node, (2) tranfering this block from the storage node buffer to a buffer in the network node across the multiprocessor interconnection network and (3) delivering the requested block to the user over the delivery medium. The critical issues in these three components of service are disk scheduling, interconnection network scheduling and delivery guarantees over the delivery medium respectively. We discuss each of these service components briefly here.

## 2.1 Disk Scheduling

Traditionally, disks have used seek optimization techniques such as SCAN or shortest seek time first (SSTF) for minimizing the arm movement in serving the requests. These techniques reduce the disk arm utilization by serving requests close to the disk arm. The request queue is ordered by the relative position of the requests on the disk surface to reduce the seek overheads. Even though these techniques utilize the disk arm efficiently, they may not be suitable for real-time environments since they do not have a notion of time or deadlines in making a scheduling decision.

In real-time systems, when requests have to be satisfied within deadlines, algorithms such as earliest deadline first, and least slack time first are used. Earliest deadline first (EDF) algorithm is shown to be optimal [3] if the service times of the requests are known in advance. However, the disk service time for a request depends on the relative position of the request from the current position of the read-write head. Moreover, due to the overheads of seek time, strict real-time scheduling of disk arm may result in excessive seek time cost and poor utilization of the disk.

New disk scheduling algorithms have been proposed recently that combine the real-time scheduling policies with the disk seek-optimizing techniques. Some of these algorithms include SCAN-EDF

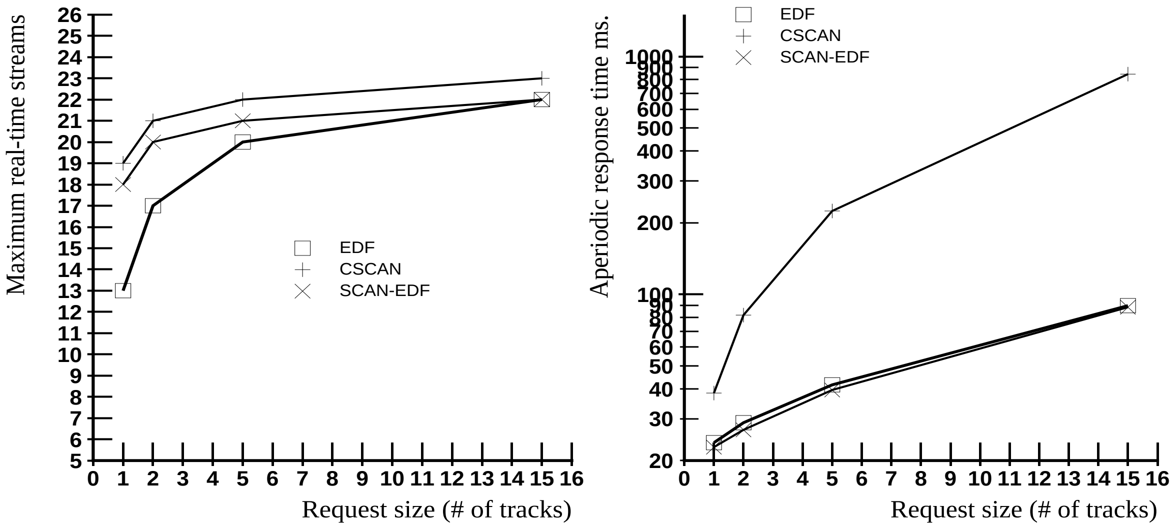

Fig. 2. Performance impact of scheduling policies and request block size.

[2] and Grouped sweeping scheduling [4]. These algorithms batch requests into rounds or groups based on request deadlines, serving requests within a round based on their disk track locations to optimize the seek time. These algorithms are shown to significantly improve real-time performance at smaller request blocks and improve non-real-time performance at larger request blocks [2]. Fig. 2 shows the impact of request block size on various scheduling algorithms.

### 2.2 Multiprocessor communication scheduling

Communication scheduling problem deals with the issue of scheduling the network resources of the multiprocessor computer system for minimizing the communication delays. If the nodes in the multiprocessor system are interconnected by a complete crossbar network, there is no communication scheduling problem since any pair of nodes in the system can communicate without a conflict in the network. Regular distribution of video data over the storage networks enables us to guarantee that if source and destination pairs are scheduled without a conflict for the transmission of the first block of data then the entire playback of that video stream will not see any source and destination conflicts [5]. Providing similar guarantees within the multiprocessor network depends on the network organization. Some interconnection networks such as Omega networks and hypercubes have been shown to exhibit properties that make it amenable to provide guarantees of no conflicts within the network. A simple solution such as a round-robin distribution of data blocks among the storage nodes suffices to provide such guarantees [5] in these networks.

The basic approach to providing guarantees of conflict-free transfers relies on time-division-multiplexing of the network resources. The network usage is divided into a number of slots. Each slot provides sufficient time to transfer a fixed block across the network under no conflicts. Each stream is assigned a fixed slot. The transfers are repeated after a fixed amount of time (a frame) to transfer the next block of each stream. The regular data distribution and the network properties ensure that if the first block of a stream is scheduled without any conflicts at the source, destination and within the network, the entire stream of blocks of that video stream can be transfered without any conflicts.

## 2.3 Other issues

In this section, we briefly touch upon a number of issues that have significant impact on the design of a video server. The intent is to give the reader a feel of the problems and to provide starting pointers for further reading.

### 2.3.1 Buffering

Buffering can make a significant impact on the videoserver design. Once, the video stream is started on the consumer's monitor, the delivery of the video stream has a real-time nature i.e., the data has to be supplied at a constant rate. However, the system can control when the first block of the stream is delivered to the consumer's monitor (latency). Buffering can be used effectively to control the latency of delivery in video-on-demand applications. The more the data is buffered, the more the stream startup latency and the more time to serve a request block at the server. In more interactive applications, long latencies cannot be tolerated. Buffering can be used to increase the request block size or to provide extra time for serving a request of a video stream (deadline extension). Both these options, larger request blocks and deadline extensions are shown to improve the number of video streams that can be supported [2].

### 2.3.2 Interactive response

Providing a VCR-like capability of pause/resume, fast-forward/reverse requires that sufficient resources be available at the server to absorb the variations in bandwidth demands due to such operations [6]. Pause operation may reduce the bandwidth demand at the server and other three operations may increase the bandwidth requirements. To support such operations, the server has to determine that these bandwidth variations do not affect the delivery of other scheduled streams. It is also possible to keep the bandwidth requirements at the server constant by serving alternate copies of video streams that have less quality encoding at the higher frame rates required by the fast-forward and reverse operations [7]. Then the server can serve the alternate copies of video data during fast-forward and reverse operations without altering the schedule of the scheduled streams.

### 2.3.3 Fault tolerance

Bandwidth considerations, load balancing and other issues favor striping data across multiple disks and nodes in a multiprocessor video server. However, this increases the probability of losing service due to a failure of one of the disks or nodes in the system. In "line of business" applications like ITV, VOD, and DVB, the video server must be highly reliable and tolerate hardware failures with minimal or no system downtime. To illustrate the importance of this, consider a large ITV system supporting 30,000 users (systems of this size are being built as of this writing [8, 9, 10]). Assuming the primary application is pay-per-view movies at $5 apiece, the system is generating approximately $75,000 per hour! This is approximately a year's pay for the service person, who undoubtedly would not collect it for long if outages were frequent or prolonged.

Multilevel encoding has been proposed to tolerate against failures in the delivery medium. In these schemes, the video data is stored in the server in two layers, a base layer and an enhancement layer. When the available bandwidth is affected due to failures, the server can continue to provide a lower quality picture to the consumer by sending only the base layer and thereby putting less load on the network. A similar idea can be used for reducing the disk overhead by protecting against the loss of the base layer only and by not providing redundancy to protect against the loss of the enhancement layer.

As opposed to a conventional file or database system, loading from tape is not a desirable way to recover from storage device failures. At the 6 Mbit/second video data rate common in most ITV systems, a two hour movie requires 5.2 GB of storage, which takes 1/2 hour to restore even from a relatively high-performance 3 Mbyte/sec tape drive. Some ITV systems in use by customer trials have 1 TB online [11], which would require 97 hours (or many tape drives!) to restore. This makes redundant storage (mirroring or RAID) a requirement.

Disk failures can be protected either by making duplicate copies of data (disk mirroring) or by parity protection (RAID techniques [12]). These fault-tolerance techniques increase the availability of data and typically can tolerate the loss of a disk without losing access to data. However, these data protection techniques incur extra load under a failure and this extra workload needs to be factored into the design for providing guarantees after a failure. To protect against the failure of a SCSI bus (bus that interconnects multiple disks to a system's memory bus), it is possible to connect the disks to two SCSI buses such that the disks may be accessible through a second bus after a failure.

These techniques can guarantee availability of data but cannot guarantee availability of the necessary I/O bandwidth for timely delivery of data after a failure. Scheduling data delivery after a component failure is a difficult problem. Much work needs to be done in this area. Overdesigning the system is one of the possible options such that even after a failure the data can be delivered in a timely fashion. Dynamic resource allocation for tolerating failures is another possible option. In certain cases, only statistical guarantees may be provided as to the capability of providing the required real-time bandwidth after a failure.

### 2.3.4 Variable bandwidth devices

Some of the current disks utilize *zone-bit recording* which makes the track capacity a variable depending on the location of the track on the disk. This results in variable data rate depending on the track location. The variable data rate makes it harder to utilize the disk bandwidht efficiently. To guarantee the real-time requirements, the minimum supportable bandwidth can be assumed to be the average deliverable bandwidth. However, this results in inefficient use of the disk bandwidth. It is possible to use the remaining bandwidth for non-real-time requests while utilizing the minimum disk bandwidth for real-time requests. Request spreading techniques [13] can be used to make sure that the requests are spread uniformly over the surface of the disk to utilize the disk at its average bandwidth.

# 3 Video Networks

The three most prominent types of digital networks used for ITV are ADSL, Hybrid Fiber-Coax, and ATM. While there are major differences between these networks, they all support both a high bandwidth video channel for sending video to the set-top box, and a lower-bandwdth control channel (usually bi-directional) for communicating user commands and other control information with the server. ADSL (Asynchronous Digital Subscriber Loop) multiplexes the downstream video channel, a bidirectional 16Kbit/sec control channel (which usually employs X.25 packet switching), and the existing analog telephone signal over the same copper twisted pair [14, 15] ADSL is designed to leverage the existing telephone company (telco) wiring infrastructore, and as such has been used in several VOD trials conducted by telcos [16, 17, 18].

ADSL is used for the connection between the customer premisis and the telephone company central switching office. The server can either be located in the central office or at some centralized metropolitan or regional center, in which case the data travels between the server and the customer's central office over the telephone company's digital trunk lines. The video bandwidth ADSL is capable of supporting depends upon the length of the subsrciber loop (i.e. the distance between the central office and the customer premisis). At present, ADSL will support 1.544 Mbit/sec (T1 bandwidth) over about three miles, which is the radius serviced by most urban central offices. MPEG compressed at T1 bandwidth can deliver VCR quality video, but is marginal for applications like sports. The choice of T1 bandwidth is convenient because it is widely used in the telco trunk system and because T1 modems are readily available to connect the video server to the digital network.

As ADSL attempts to leverage the existing telephony infrastructure, Hybrid Fiber-Coax (HFC, [19]) leverages the existing cable TV plant, and as such is popular for ITV systems deployed by cable service providers. Standard cable plants deliver some number of analog TV channels (typically 70 channels using a total bandwidth of 450 MHz), which are fanned out to all subscribers. The cable system is structured as a large tree, with repeaters at each of the nodes and subscriber cable decoders at the leaves. Seventy channels is barely a sufficient number for broadcast, and is not nearly sufficient for VOD or ITV. The basic idea behind HFC is to subdivide the tree into smaller subtrees (for example, one subtree per neighborhood), and feed each subtree independently by fiber from the cable company head end. Some number of channels in this subtree are still reserved for broadcast, and the remainder are available for interactive services. A segment of the cable's frequency range (typically 5-30 Mhz) is reserved for upstream control signalling, and LAN-like packet switching is used on this upstream channel.

In an HFC system, each user of interactive services is assigned a channel for the period during which he is using the system. The server sends a command over the control channel to tell the STB which channel to select, and subsequently video to the STB over this channel. For security, STBs can prevent interactive channels from being selected manually.

In some early trials, a subset of the analog TV channels were reserved for ITV [20]. MPEG video from the server was converted to analog at the cable head end and multiplexed onto fiber. Since only a small number of channels (those not used for broadcast) on each subtree are available for ITV, the area covered by a subtree is small, and therefore a completely analog system is expensive.

Recently, new techniques (VSB, QAM) have been developed that allow sending high-bandwidth digital data over analog TV channels. QAM-64, for example, allows 27 Mbit/sec of digital data to be carried by one 6 MHz analog TV channel. This payload can carry one high-bandwidth program (e.g. HDTV), but is normally used to carry an MPEG-2 transport stream with a number of lower-bandwidth programs. For example, QAM-64 allows four 6 Mbit/sec MPEG-2 programs to be carried on a single 6 MHz analog TV channel, allowing a 70-channel cable TV plant to carry 280 channels. Higher-bandwidth fiber-based systems are being deployed that support a downstream bandwidth of 750 MHz, which supports over 400 6 Mbit/sec MPEG programs. This is suitable for carrying a full complement of broadcast channels as well as a sufficient number of interactive channels for a neighborhood of 1000 households.

The downstream video path of a HFC video server is somewhat more complicated than its ADSL counterpart. The server sends individual MPEG programs to an MPEG multiplexor, which combines the independent programs into a single MPEG transport stream. The transport stream is then converted (e.g. by a QAM modulator) to a 6 MHz analog signal and then multiplexed onto one of the cable system's analog channels. The STB selects its MPEG program under control of the server form one of the MPEG transport streams.

While ATM has not been deployed at nearly the rate anticipated a few years ago, it is still widely regarded as the future architecture for ITV. ATM has a number of strong advantages: it can support the high bandwidth required for video, it can multiplex video and control data over the same connection, and it allows bandwidth to be reserved over an end-to-end connection. A number of alternatives exist for transporting video over ATM. ATM defines a standard (AAL1) for continuous bit rate traffic, and another standard (AAL5) for packet-based data. Perhaps surprisingly, most ITV systems are using the packet-based approach. Even using AAL5, several alternatives exist: the ATM fabric can be used as a fast IP network, or a specialized video streaming protocol can be used [21]. Most ATM-based ITV sysems implement the ATM Forum standard; examples of IP-based systems are discussed in a later section. Time-Warner's "FSN" ITV system, deployed in Orlando, Florida [22], uses a combination of ATM and HFC. The server sends multiple MPEG program streams via ATM to a QAM modulator. From there, they are transmitted by fiber to the customer STB as discussed above. Here, ATM is used primarily as a head-end switching fabric rather than as a distribution mechanism.

## 3.1 ITV Server Architecture

Although the ITV server is often described as if it were a homogeneous black box, it is composed of a number of logically and usually physically independent pieces. Following is a description of an example ITV server built by IBM for the Hong Kong Telecom VOD Trial [23]. This system uses ADSL to distribute video, and is similar to ITV systems being deployed by some U.S. Telcos. Fig. 3. shows the trial setup used by IBM in Hong Kong. The ITV system as a whole comprises an open system designed to allow an individual customer to connect to any of a variety of video servers. To this end, the network contains a component called the Level 1 Gateway, which serves as an intermediary between the customer and a corresponding Level 2 Gateway in each video server. This gateway architecture is the so-called video dial-tone system mandated by the FCC for ITV systems that use regulated common carriers. It is analogous to the standard telephony system, which allows a customer to connect to any desired long-distance carrier. The STB contains a control port, used to communicate with the video server and the Level 1 Gateway over the control channel using X.25, and a data port, over which the video server sends video and graphical images and downloads code to the STB at DS-1 data rate.

The video server contains an X.25 control port, used to communicate with the Level 1 Gateway and STB control ports. The video server's control interface to the Level 1 Gateway is called the Level 2 Gateway. This "gateway" architecture is what allows video servers from multiple service providers to connect to the system in an open manner. The video server also contains a number of DS-1 rate data ports sufficient to support the maximum expected number of active viewers.

**Level 1 Gateway.** The Level 1 Gateway (GWL1) provides the Video Dial Tone for the set tops. The GWL1 processes requests for service from the set tops. If the set top is a valid subscriber, the GWL1 responds with a list of available video servers. Alternatively, and appropriate for many trials, the set top can be programmed to automatically select a default server in the initial request message. The GWL1 also establishes the high-speed data connection between the server and the STB. This is done in response to a request by the GWL2, specifying the server data port address and the STB identifier. The GWL1 maintains tables containing the data port address for each STB, and uses this table to issue commands to the data network switching equipment to connect

the server port and the STB.

**Level 2 Gateway.** The Level 2 Gateway (GWL2) in the video server provides a standardized, open interface between the server and the network. As specified in [BA93b], the GWL2 performs the following functions:

- Connection to GWL1. Upon start-up, the GWL2 initiates an X.25 connection to the GWL1 for the exchange of control information.
- Processing STB connection requests from GWL1. When the GWL1 receives a request from an STB to connect to a server, it sends a connection request to that server's GWL2. This request contains the STB's unique ID, its brand and version, and its network address.
- Initiating control and data connections to the STB. The GWL2 allocates an unused data output port, and sends a request to the GWL1 to establish a connection between the data output port and the data input port of the STB, and finally establishes an X.25 control connection to the STB.

The GWL2 then downloads the version of Set Top Enabling Code appropriate to that brand and version of STB. The STEC software provides the set top with the means to process the subsequent commands it will receive.

## 4 Network Multimedia

During the last several years the cost of compute power, storage devices, and storage media has dropped readically in the personal computer marketplace. This has led to the proliferation of interactive multimedia applications and games for the PC. To date, most multimedia applications run on stand-alone PCs, with digitized video and audio coming from local hard disks and CD-ROMs. Increasingly, there has been a demand for file servers that support capture, storage and playback of multimedia data. The reasons for this are identical to the ones that motivate the use of file servers for conventional data: sharing, security, data integrity, and centralized administration.

### 4.1 Servers for Network Multimedia

Although Interactive TV is the glamor application for video servers, video playback over a LAN is the bread and butter of the industry. A LAN multimedia system includes one or more servers, some number of multimedia client workstations, and a video-capable LAN. Applications for LAN multimedia include training videos, information kiosks, desktop news, and digital libraries.

The client workstation software architecture has much to do with the design of the multimedia LAN server. There are two basic client architectures, depending upon whether or not the manufacturer-supplied driver software provided with the video codec is used.

Normally, the video codec (e.g. MPEG card, Indeo software codec) comes packaged with driver software that allows video to be played through the operating system multimedia support (e.g. the Windows Media Player or MCI programming interface). The loop to read a buffer of video data and present it on the display is implemented in this driver, which is a "black box" executable file supplied by the decoder manufacturer. Since this driver reads data through the file system interface, the server must make video available through that interface. Fig. 4. shows a typical multimedia LAN server.

Most client platforms (Windows, Macintosh, Unix) support the notion of an installable file system (IFS) which the multimedia LAN server can use to provide video data. However, the performance characteristics of the multimedia IFS driver must be equivalent to local disk or CD-ROM. In particular, any read-ahead or extra buffering to smooth out variations in the network transfer rate must be implemented in the IFS driver, not in the video codec device driver (which is unaware that video is coming from the server as opposed to from local disk).

This file-based approach has the advantage of allowing the server to support almost any video codec. Furthermore, applications that use the operating system media player or multimedia programming interfaces (most multimedia applications fall into this class) need not be modified to use the server. For example, such servers allow copying a multimedia CD-ROM to the server, after which it can transparently be used by any number of clients simultaneously.

The server can either implement its own proprietary installable file system, or can use one designed to support a standard Network Operating System protocol (e.g. Netware, LAN Manager, or NFS). A proprietary file system allows using more aggressive readahead and buffering or a network transport protocol tuned for video. However, a number of commercially available servers use standard NOS IFS drivers (or NOS Clients). This has obvious advantages: it allows the server

to support a large variety of platforms and makes it unnecessary for the user to install special software to use the server. Several commerical servers have been built using standard NOS Clients; examples include IBM OS/2 LAN Server Ultimedia [24], Panasonic Video NFS [25], and IBM Multimedia LAN Server for AIX [26].

Contrary to popular belief, standard NOS network protocols work acceptably for video over a suitable LAN, that is, one with sufficiently high throughput, low latency, and low error rate. The need for throughput is obvious; low latency is required to prevent buffer starvation with the limited amount of readahed done by most standard NOS clients. For example, the NFS protocol supports a maximum 8K read size. Application (e.g.. codec driver) reads requests larger than 8K are broken up by the NOS Client and sent to the server as multiple 8K requests. If the NOS Client sends these requests synchronously (i.e. one at a time), the time to complete a request must average less than 54 msec. to maintain a data rate of 1.2 Mbits/sec (the data rate for most CD-ROM MPEG). By supporting readahead (sending successive requests before previous ones complete), a NOS Client can relax this latency requirement somewhat. Most NOS cleints support readahead, but some older ones do not. The final requirement for low error rate stems from the fact that in most NOS clients, error recovery is driven by timeouts. These timeouts are typically short enough to allow the occasional dropped packet without a visible interruption in video, but bursts of errors will cause video to degrade.

Modern technology allows configuring LANs to support video in a cost-effective manner. Switched Ethernet, FDDI, and ATM LAN Emulation are all suitable for video, and with high-performance routers can be scaled up to support a large building or campus.

A number of Multimedia LAN Server vendors have taken the approach of implementing proprietary streaming protocols to transport video over LANs. Typically this forces the server vendor to supply special driver software for each video codec it wishes to support. This usually restricts such servers to a limited number of codecs and increases the time required to support new ones. However, a proprietary streaming protocol can in principle be designed to work better with a less-than-perfect network. Starlight Networks' Star Works and its MTP protocol is a successful example of this approach [27, 28].

## 4.2 Internet Video Servers

The recent explosion of interest in the internet and the Worldwide Web have resulted in a number of attempts to support video over long distances using the internet. These range from modest attempts to support low-resolution video over dial-up and ISDN connections, to more ambitions attempts to transport high-resolution video over the MBONE WAN.

The earliest attempts to support video and audio over the Worldwide Web used store and forward. Using this technique, the video file is downloaded to the client workstation in its entirety, after which the operating system media player is launched to play the file. For all but the shortest videos, the delay to download before viewing is unacceptable.

Recently a number of systems have become available that play video and audio in realtime over the internet. Most are implemented as a Web browser external viewer or plug in (the distinction is not important to this discussion), and most work more or less the same way. When the user clicks on a link to video (or audio) on a web page, the browser loads the contents of the file pointed to by the link and launches the viewer (or plug-in). This file, instead of containing the video data itself, contains a URL pointer to the data, often on a separate, dedicated video server. The viewer starts transferring data from the server, buffering it in RAM until it has a sufficient amount to begin playback. Video is decoded in software and displayed in a window; accompanying audio is played from the data buffer using the low-level audio subsystem programming interfaces (e.g. waveOutWrite on Windows).

In contrast to the file-based approach, this type of viewer usually does not take advantage of special hardware (e.g. MPEG cards) that might be on the system, or supports a limited number of cards using special drivers. Given the low video data rates possible over the internet at present, this is usually not a serious consideration. On the other hand, rather than using a conventional network file system protocol (e.g. NFS) or even a conventional streaming protocol (e.g. TCP), these viewers usually implement special-purpose network transport protocols (layered on UDP) that are designed specifically for audio or video. Some additionally implement proprietary compression algorithms tailored to the internet's low data rate and high error rate. Commercial products implementing proprietary transport protocols and/or compression algorithms include VDO Live [29], Real Audio, Xing StreamWorks [30], and InSoft [31].

The research community and standards organizations have also been active in transporting multimedia over the internet. RTP defines an encapsulation of any of a number of audio or video data formats into UDP datagrams that allows it to be streamed over the internet. Rather than retransmitting lost packets as does TCP, RTP timestamps packets to allow allow missing packets to be detected. RTP contains no flow-control mechanism, and leaves the task of recovering from lost packets to the video/audio codec. A number of commercial products are based on RTP (Precept FlashWare, Netscape LiveMedia), and several other products have announced intentions to suport it (Xing, Real Audio, VDO Live).

At the high end, research is ongoing to stream video at high data rates over high-speed backbone networks such as the MBONE. The *vic* video conferencing system supports real-time video multicast using RTP [32]. While the original version of vic supported only live video, Argonne National Labs has extended vic to support playback of stored video at high bandwidth (6 Mbits/sec) via RTP from IBM's Tiger Shark video server [33] on an IBM SP-2 parallel computer [34]. Research is also underway to develop still better protocols. One such protocol, PET [35], encodes MPEG data to add sufficient redundancy to tolerate high levels of packet loss while maintaining good picture quality. PET assigns priorities to each component of the MPEG data stream (headers, I-frames, B-frames, and B-frames) and uses more redundancy for higher-priority data. With only a modest space overhead, PET greatly increases MPEG's ability to tolerate data transmission errors. VTP [36] is an extension to RTP to provide a form of flow control by asking the server to slow down or speed up as network conditions dictate, and a form of error recovery by allowing the client to "demand resend" dropped video frames.

## References

[1] A. Chervenak. Tertiary storage: an evaluation of new applications. *Ph.D Thesis, Univ. of Calif., Berkeley*, 1994.

[2] A. L. Narasimha Reddy and Jim Wyllie. Disk scheduling in a multimedia I/O system. *Proc. of ACM Multimedia Conf.*, Aug. 1992.

[3] C. L. Liu and J. W. Layland. Scheduling algorithms for multiprogramming in a hard real-time environment. *Journal of ACM*, pages 46–61, 1973.

[4] P. S. Yu, M. S. Chen, and D. D. Kandlur. Grouped sweeping scheduling for dasd-based multimedia storage management. *Multimedia Systems*, 1:99–109, 1993.

[5] A. L. N. Reddy. Scheduling and data distribution in a multiprocessor video server. *Proc. of Int. Conf. on Multimedia Computing and Systems*, pages 256–263, 1995.

[6] K. D. Jayanta, J. D. Salehi, J. F. Kurose, and D. Towsley. Providing vcr capabilities in large-scale video server. *Proc. of ACM Multimedia Conf.*, pages 25–32, Oct. 1994.

[7] M. S. Chen, D. Kandlur, and P. S. Yu. Support for fully interactive playout in a disk-array-based video server. *Proc. of ACM Multimedia Conf.*, pages 391–398, Oct. 1994.

[8] *Wall Street Journal*, Nov. 10, 1995.

[9] Hong kong to get world's first multimedia service. *Computergram International*, Nov 8, 1995.

[10] Nec to supply set-top boxes to hong kong telecom ims. *Business Wire*, November 14, 1995.

[11] Basic plan for the multimedia experiments at the tokyo metropolitan waterfront subcenter multimedia experiments. *http://www.tokyo-teleport.co.jp/english/atms/atmsj201.htm.*

[12] D. A. Patterson, G. Gibson, and R. H. Katz. A case for redundant arrays of inexpensive disks (RAID). *ACM SIGMOD Conference*, June 1988.

[13] Y. Birk. Track pairing: A novel data layout for VOD servers with multi-zone recording disks. *Proc. of IEEE Conf. on Mult. Comput. and Systems*, 1995.

[14] Signalling specification for video dial tone. *Bell Atlantic Inc., TR-72540*, Dec. 1993.

[15] Adsl video dial tone network interface requirements. *TR-72535, Bell Atlantic, Inc.*, Dec. 1993.

[16] Bell atlantic, ibm announce agreement for video on demand server. *World News Today*, Jan 8, 1993.

[17] Larry Plumb. Bell atlantic demonstrates video on demand over existing telephone network. *Bell Atlantic press release*, Jun. 14, 1993.

[18] Alan Patterson. Hong kong telecom, ibm map video effort. *Electronic Engineering Times*, Aug. 1, 1994.

[19] Walter S Ciciora. Cable television in the united states - an overview. *Cable Television Labs Inc.*, 1995.

[20] Cox to launch interactive tests in omaha. *United Press International*, Dec. 1, 1993.

[21] ATM Forum. Document 95-0012r5. 1995.

[22] Time warner cable completes first phase of full service network. *Time Warner press release*, Dec. 16, 1993.

[23] R. Haskin and F. Stein. A system for the deliver of interactive television programming. *Proc. of IEEE 1995 Spring COMPCON, San Francisco, CA*, pages 209–216, Mar. 1995.

[24] Ibm lan server ultimedia - multimedia for os/2, dos, and windows clients. *IBM Document 96F8520*, Nov. 1993.

[25] Matsushita develops "video server" technology for use on standard computer networks. *Matsushita Electric Industrial Co. Ltd. Press Release*, Mar. 22, 1995.

[26] Ibm multimedia server for aix, version 1.1. *IBM Document 5967-213*, Feb. 20, 1996.

[27] The challenges of networking video applications. *Starlight Networks Technical Report, Starlight Networks Inc., Mountain View, CA*, April 22, 1996.

[28] Effect of video on lan data traffic. *Starlight Networks Technical Report, Starlight Networks Inc., Mountain View, CA*, April 22, 1996.

[29] Vdolive will enable motion video on the internet. *VDOnet Corp. Press Release (http://www.vdolive.com)*, Oct. 27, 1995.

[30] Xing streamworks squeezes video through 14.4 and 28.8 dial-up modems. *Business Wire, Arroyo Grande, CA*, Oct. 17, 1995.

[31] Insoft targets internet real time multimedia users and developers with first interactive collaborative environment. *Insoft Inc. Press Release, Mechanicsburg, PA*, Jan 8, 1996.

[32] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. Rtp: A transport protocol for real-time applications. *Internet RFC 1889, Internet Engineering Task Force*, January 1996.

[33] R. Haskin. The shark continuous media file server. *Proc. of IEEE 1993 Spring COMPCON, San Francisco, CA*, pages 12–17, Feb. 1993.

[34] The medium is the message: Ibm, argonne national lab host first bi-coastal "internews" conference. *IBM Corp. Press Release IBM US 490, San Diego, CA*, Dec. 6 1995.

[35] A. Albanese, J. Bloemer, J. Edmonds, and M. Luby. Priority encoding transmission. *Int. Comp. Sci. Inst. Tech. Rep. No. TR-94-039*, August 1994.

[36] Z. Chen, S.M. Tan, R. H. Campbell, and Y. Li. Real time video and audio in the world wide web. *Proc. of the 4th Int. World Wide Web Conference*, Dec. 11, 1995.