Quality of Service based Retrieval Strategy for Distributed Video on Demand on Multiple Servers

A

Thesis

Submitted in

Partial fulfillment of the requirement for the degree of

Master of Technology in Computer Science

By

Niranjan Panigrahi

Under the guidance of

Prof. Bibhudatta Sahoo

Department of CSE

Department of Computer Science & Engineering
NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA
ROURKELA-769008 (ORISSA)

National Institute of Technology Rourkela
Rourkela-769008 (Orissa)

Certificate

This is to certify that the work in this Thesis Report entitled "***Quality of Service based Retrieval Strategy for Distributed Video on Demand on Multiple Servers***" by *Niranjan Panigrahi* has been carried out under my supervision in partial fulfillment of the requirements for the degree of ***Master of Technology*** in Computer Science during the session 2007-2009 in the Department of Computer Science and Engineering, National Institute of Technology , Rourkela, and this work has not been submitted elsewhere for a degree.

Place: Rourkela (Prof. Bibhudatta Sahoo)

Date: May 26,2009 Department of CSE

Acknowledgements

No thesis is created entirely by an individual, many people have helped to create this thesis and each of their contribution has been valuable. I express my sincere gratitude to my thesis supervisor, *Bibhudatta Sahoo*, Sr. *Lecturer, CSE,* for his kind and able guidance for the completion of the thesis work. His consistent support and intellectual guidance made me energize and innovate new ideas.

I am grateful to *Dr.B.Majhi*, *Professor and Head, CSE* for his excellent support during my work. Thank are due to all my classmates for their love and support.

Last, but not least I would like to thank all professors and lecturers, and members of the department of Computer Science, Engineering, N.I.T Rourkela for their generous help in various ways for the completion of this thesis.

(Niranjan Panigrahi)
Roll No-207CS115,
M.Tech. (Comp. Sc.),
Session- 2007-2009

Contents

Abstract

The recent advances and development of inexpensive computers and high speed networking technology have enabled the Video on Demand (VoD) application to connect to shared-computing servers, replacing the traditional computing environments where each application was having its own dedicated computing hardware. The VoD application enables the viewer to select, from a list of video files, his favorite video file and watch its reproduction at will. Early video on demand applications were based on single video server where video streams are initiated from a single server, then with the increase in the number of the clients who became interested in VoD services, the focus became on Distributed VoD architectures (DVoD) where the context of distribution may be distributed system components, distributed streaming servers, distributed media content etc.The VoD server must handle several issues in order to be able to present a successful service. It has to receive the clients' requests and analyze them, calculate the necessary resources for each request, and decide whether a request can be admitted or not. Once the request is admitted, the server must schedule the request, retrieve the required video data and send the video data in a timely manner so that the client does not suffer data starvation in his buffer during the video reproduction. So, the overall objective of a VoD service provider is to provide a better Quality of Service (QoS). Some issues related to QoS are-efficient use of bandwidth, providing better throughput etc.One of the important issues is to retrieve the video data from the servers in minimum time and to start the playback of the video at client side with a minimum waiting time. The overall time elapsed in retrieving the video data and starting the playback is known as access time. The thesis presents an efficient retrieval strategy for a distributed VoD environment where the basic objective is to minimize the access time by maintaining the presentation continuity at the client side. We have neglected some of the network parameters which may affect the access time, by assuming a high speed network between the servers and the client. The performance of the strategy has been analyzed and is compared with the referred PAR (Play After Retrieval) strategy. Further, the strategy is also analyzed under availability condition which is a more realistic approach.

List of Figures

List of Tables

Abbreviations

VC	Video Conference
VCR	Video Cassette Recorder
NTSC	National Television System Committee
MCU	Multipoint Control Unit
ATM	Asynchronous Transfer Mode
OC	Optical Carrier
SDV	Switched Digital Video
MMDS	Multipoint Microwave Distribution System
VoD	Video on Demand
MoD	Movie on Demand
QoS	Quality of Service
CoS	Class of Services
PAR	Play After Retrieval
MSR	Multiple Server Retrieval
DFG	Directed Flow Graph
MPEG	Moving Pictures Expert Group
ITU	International Telecommunication Union
HDTV	High Definition TV
PPV	Pay Per View
QVoD	Quasi Video on Demand
NVoD	Near Video on Demand
TVoD	True Video on Demand
MTTF	Mean Time To Failure
MTBF	Mean Time Between Failure
MTTR	Mean Time To Repair

Chapter 1
Introduction

Introduction Chapter 1

1.1 Introduction

Multimedia communication applications have the potential to improve the level of human-computer interaction by providing audio, video and other media along with traditional media such as text, graphics and images. Advances in research and development and the convergence of computers, telecommunications, consumer electronics and entertainment are making it possible to see some practical systems in the distributed multimedia area. Typical applications include Video Conference (VC), IP telephony, Multimedia Mail, Multimedia Mall, Digital Libraries, and Video on Demand (VoD)[14]. These applications need to acquire their requested service levels and qualities, varying largely one from another. For example, an IP telephony application requires voice signals arriving within a tolerated delay variance (jitter); a video player requires a bandwidth guarantees to convey its images smoothly; an online mall requires all data secured without loss and a good quality video-on-demand system requires [6]**:** a jitter-free video display, synchronized video and audio, a video playback rate of at least 30 frames per second and complete user control over viewing.

But none of the above applications has yet been identified as a *killer* application or a *killer suite* of applications by the multimedia information industries, to produce significantly greater profits and revenues. Many of the companies in the area of telecommunications, computers, entertainment and multimedia content production are merging, and are conducting many trials to predict technical feasibility and market feasibility. Such a trial study has shown that some problems are technical and some are related to the consumers' satisfaction [14]. From consumer side, it could involve consumers' acceptance of emerging technology, willingness to pay for these new applications etc.

High speed networking technology coupled with the availability of sophisticated multimedia computing and communication devices makes network based multimedia services, such as video/movie-on-demand (VoD) viable. Last decade has witnessed a tremendous advancement in theoretical as well as practical realizations of VoD systems. In fact, commercial VoD services with complete video cassette recorder (VCR) functions have appeared. However, owing to ever increasing user demands, when the user access rates increase, several issues need to be tackled, e.g., high block rate, long startup delay, glitch (i.e. service interruption), jitter(e.g., frame losing) etc.[2].

More precisely these distributed multimedia applications need to meet the underlying QoS requirements. From users' perspective, the QoS requirements are always subjective in nature. So, these parameters have to be mapped to the underlying system to find out the appropriate objective (i.e quantitative) parameters and must be optimized to provide a both technically and economically sound application.

1.1.1 Distributed Multimedia Requirements

This section briefly discusses the platforms and technologies requirements for distributed multimedia applications and finally highlights on the specific requirements of VoD service .The requirements [14] are summarized as follows:

(i) Application Programming Interfaces: We need portable user interfaces, smart agents and conference management.

(ii) Audio Quality: Using conventional speaker phones (usually half duplex) with full-motion video, the result is disturbing; the participants can see another person's lips moving but cannot hear them **.**A full-duplex echo canceling speaker phone is a good choice.

(iii) Video Quality: Quality is often inadequate, and it is much less than the broadcast quality. Parallax-free viewing (i.e. direct eye contact) and proper face lighting, along with NTSC image quality, are preferred. Full-size faces **as** opposed to talking heads are preferred. Higher quality cameras could be used as a tradeoff to compensate for blinding the user with extra light.

(iv)Multimedia Object Technology: A database should now include audio, video and other media objects. Standard software with object data bases cannot meet large-scale VoD requirements due to performance, real time constraints and object output controls.

(v) Multimedia bridging: Although some manufacturers have Multipoint Control Unit (MCU) for conference bridging on the market, the costs are still prohibitive for a small time provider of services.

***(vi)** Standards:* Many of the industry providers are overzealous when they set high levels of expectation for quick deployment and successful penetration of applications. Simply investing in digitizing video and audio information is not the solution when there are multiple standards and proprietary methods that exist and compete with each other. Open standards give buyers a choice of vendors, offers a promise of compatibility, and gives an assurance that equipment will not quickly become obsolete. For manufacturers, this customer confidence means a larger market, leading to larger volumes, lower prices, and a greater variety of available products.

*(vii) Asynchronous Transfer Mode (ATM***):*** ATM technology provides fast packet transport and switching and multiplexing of packets. It supports synchronous and isochronous media, large bandwidth, flexible dynamic bandwidth, and supports standards. It supports speeds from OC-3(155.50 Mbps) to OC-12(622.08Mbps) speeds.

(viii) Hardware: There is need for a single VLSI chip for compression, decompression, CPU, Asynchronous Transfer Mode (ATM) segmentation and reassemble. Storage technology needs revitalization in improving the speeds and be able to have mass storage. Development of Super dense optical storage from Sony & Philips, Toshiba and Time Warner is useful.

*(ix) Cable Modems**:*** More than 30 million US households have PCs at home. Broadband CATV networks pass more than 90% of US households. This network can be used to transport multimedia traffic by upgrading to hybrid fiber coax architectures. The key benefit of cable modems is that they are ten to hundred times faster than dial-up modems or ISDN. CATV is universally available and inexpensive. Companies such **as** Lucent Technologies, GI, ADC, NewBridge, Zenith and LANcity are developing cable modems.

(x) Loop and access sub-network: The ADSL technology is promising **as** it can deliver up to 6 Mbps using the currently available copper. The other technologies could be Switched Digital Video (SDV) and Multipoint Microwave Distribution System (MMDS). Multimedia systems will impact computer systems, storage memory in terms of capacity, access time and transfer rate, interconnections (for example, bus bandwidth),

processing power **(as** software codecs have to be processed), operating system, and network bandwidth. VOD could take up 173Mbps (i.e. 30frames/second x (480 x 525) pixels/frame x 24 bits/pixel) while video conferencing can take up to 69.6 Mbps (i.e. 30 frames per sec x (288 x 352) pixels/frame x 24 bits/pixel). Technological advancement in digital electronics and fiber optic communications are making more functions economical and simpler to use. The other contributing factor is the application software that is easy, and it should compel the user to continue using it. Creating multimedia applications that are easy and interesting requires *inspiration* and *perspiration***.** Creating such an application will result in millions of users contributing to the final product.

Now talking on the requirements of VOD service, it is characterized by asymmetric information flow as real-time signaling information transmitted to the head-end is less than what is transmitted to the set top box. A VOD database of compressed video requires standards such as MPEG-2. At a video rate of 3 Mbps, an average two-hour movie can occupy 2.7 GB in disk storage. A video server with 500 on-line titles would therefore need 1.35 terabytes of storage. Many video server manufacturers have chosen ATM to transport audio and video at 2.5 GBps. A significant requirement on video server is that its output data. rate should be 400 MB/**s,** and its storage will be 1.5TB of on-line disk capacity with 6TB of off-line archival tape. Some of the broad band transport requirements on *One-way end-to-end delay, End-to-End delay jitter, differential delay, response time, Intra-media synchronization* and *Inter-media synchronization* need to be satisfied.

1.1.2Distributed Multimedia Challenges

Efficient multimedia systems are not just simple extensions of conventional computing systems. Delivery of *time sensitive* information, such as voice and video, and handling of large volumes of data require special considerations to produce successful widespread applications. Consequently, all system components, e.g., storage, operating systems, and networks, must support these additional requirements. This results in very different architectural and design decisions in all subsystems and their services. More specifically, some of the important issues [14] that have to be addressed when designing a multimedia system are:

(i) *Multimedia storage and speed:* Audio and video require larger capacities for storage and transport in comparison to text, graphics, and image. Typically, the computer speed requirements for text may be at 28.8kbps, speak and listen to audio a t 64 Kbps, and watch video at more than 1.2 Mbps.

(ii) Proprietary environments: Many of the development environments and playback or runtime environments are proprietary, making it hard to port or run on different computer systems. However, Microware Systems Corporation has developed a product called Digital Audio/Video Interactive Decoder (DAVID**)** which is based on OS-9. DAVID is a system software that provides a standard, highly functional, and cost-effective interactive multimedia software environment for the consumer market. Transporting across the network requires standard based data formats, which are slow in developing. Today there is tight coupling of data formats and Application Programming Interfaces (API) with specific devices.

(iii) Client Server models: The client and server are either incompatible or they are specific to the applications environment, and hence it is difficult to port them on another system. Client-server paradigm needs refinements to enable remote procedure calls to work effectively. Even standard based protocols like HTML have extensions that are only implemented by certain vendors.

(iv)Legal Implications: Who is held responsible for the multimedia information content distributed by the providers of service to users? Is it the company providing the content, or the company providing the technology to be held responsible for any litigation?

(v) Users ability to pay: Users are not interested in paying more than 10 to 30% of the current rates [2].

(vi) Creation of Multimedia Content: Creation of a good content demands creativity and effort. Are users willing to pay extra for this effort?

(vii) Information Ownership: When Multimedia Information is transmitted to television / PC, can a user copy the information, modify and re-use this information? Consider a multimedia book that has text, graphics, image, audio and video to make a book's content clearer and more enjoyable. Apart from the typical navigational capabilities available through different media, a network interface provides further details about any desired part of the book's material. In addition, the user could copy and paste parts of this information to become the author of a sequel, or partial co-author of a new edition of the book! When using VOD application, the user has similar capabilities to produce new movie clips from the original one. The multimedia application has made users the drivers,

by transforming the readers into authors, moviegoers to producers and directors, and program users into programmers. Do we need new rules for Information Ownership and suppress user's creativity? Will millions of amateurs, who create and distribute modified content through distributed multimedia, be significantly more valuable than the best that today's far fewer specialists can create?

(viii) Networked Multimedia: For multimedia applications such as VOD and VC, the transmission bandwidth requirements are high, making it difficult to receive multimedia information over existing modems. The network today treats individual streams **as** completely independent, and unrelated, and thus separate connections need to be done. Therefore, applications demanding multiparty connectivity require a new signaling paradigm to be implemented through efforts of ITU-T (H.323, H.320, H.225 ...) as multimedia streams are to be
synchronized.

(ix) Endpoints are treated unintelligent: Most of today's endpoints (Personal Computers (PCs)) are intelligent, yet telecommunication networks continue to treat them as unintelligent.

(x) *User Interface:* How do we incorporate new media into the human-computer-interaction paradigm? This process has to be a natural interaction. Will the research and developments in speech recognition, synthesis and analysis help us design better user interfaces? The user interface plays an important role in making or breaking a product or service's acceptance by the user.
(xi) Time for distribution: Today, getting a book published entails a large amount of delay, as does getting a movie into theaters or television broadcast. Distributed multimedia applications will greatly reduce this delay in pushing multimedia information, since they are already in digital form and can be transported over digital networks. This is useful to the users only if they can extract any portion of information that they need.

Sl. no	Authors	Approaches	Type of retrieval	Performance parameters
1.	Z. Wang and R.K.Guha [12]	Bandwidth weighted partition (BWP)	Server level	Server bandwidth, storage capacity
2.	Z. Wang and R.K.Guha [12]	Popularity based(PB)	Server level	Server bandwidth, storage capacity
3.	C. P Low[13]	BFSRA using random duplicate assignment	Disk level	Response time, disk load
4.	J. Korst[10]	Max-flow selection algorithm	Disk level	Retrieval time, disk load
5.	Aerts et al.[10]	Retrieval using random chain Declustering & Circulant graph	Disk level	Retrieval time, disk load
6.	P. Sumari et al.[8]	Storage & retrieval using residual theory rule	Disk level	Avg. waiting time, No. of streams
7.	H. Yu et al.[7]	Video server selection algorithm	Server level	Delay
8.	L. Chen and B. Veeravalli[3]	Play While Retrieval strategy	Server level	Access time, Presentation continuity
9.	B. Veeravalli and G. Barlas[1]	Play after Retrieval strategy	Server level	Access time
10.	Y.W. Leung et al.[4]	Assignment using goal programming & problem relaxation	Server level	Server load, Blocking probability
11.	E. Biersack et al.[27]	GCDL retrieval with VBR streams	Disk level	Buffer size, start-up latency

Table 1.1 Retrieval Approaches for VoD

1.2 Related Work

A central design issue of VoD service is in addressing how to store, retrieve and transmit compressed-format videos (e.g., movies) from the server sites to the clients using the underlying network without under-utilizing the resources (e.g., storage of disk and memory, bandwidth of disk and network, etc.). In detail, common focus includes, scheduling of multimedia streams for minimizing access times in broadcasting [1], network structure optimization for minimizing the transmission cost, disk scheduling for minimizing access times[8][9][10][11], while maximizing the number of continuous streams and minimizing the buffer space requirement , movie placement strategy in a disk array for minimizing the number of disks [12][13], storage hierarchy optimization for minimizing the storage cost [6], movie buffer caching strategy for effectively utilizing the memory and reducing disk I/O overheads, and dynamic network resource allocation for improving transmission rates with low jitter variation in media streams [7]. Among these researches, the design of efficient strategies to retrieve multimedia documents such as video clips, movies (short and long durations), etc, is a very important research area. The table 1.1 summarizes the various approaches followed by different authors in the area of retrieval.

From the literature study, the retrieval schemes can be broadly classified into two categories: (i) Disk level retrieval schemes[3][4][5][6][11] to efficiently allocate and migrate the multimedia data between different storage devices in order to provide smaller user request latency, load balancing and large system throughput and (ii) server level retrieval schemes[1][2][7][8][9][10] which see only online multimedia data i.e. data ready to be delivered to the clients when requests arrive. Our approach is based on server level.

1.3 Motivation

In most of the existing literature, a VoD system is usually conceived as a system having a single video server. However, with an increase in the user access rates, the network bandwidth becomes a natural bottleneck, especially for a network based multimedia service. A decentralized (distributed) approach would rather handle this increased access rates more elegantly. This is typical of a requirement on Internet like networks, where access rates may be significantly large. We assume that the transmission bandwidth is not less than the retrieval bandwidth of servers and neglect the transmission delays for the ease of analysis.

VOD service is usually open to many users and so the server system should be able to serve multiple users concurrently. To serve each user, the server system retrieves and delivers video at the video playback rate (e.g., 1.5 Mb/s for MPEG video). Consequently, the server system should have a large capacity to handle a large volume of bits. To satisfy this requirement, the server system can adopt one of two configurations [4].

1) *Centralized server configuration* [Fig1.1(a).)]—The server system uses a high-end computer (e.g., a multiprocessor computer) as a video server.

2) *Distributed server configuration* [Fig.1.1(b)]—The server system uses multiple low-end computers (e.g., personal computers) as video servers. Each server stores some of the movies, and it can serve multiple users concurrently. Overall, the server system can serve many users concurrently. Serpanous and Bouloutas [5] made a comprehensive comparison between these two configurations. In particular, the distributed server configuration is attractive in three aspects.

1) *Good scalability*—The system can easily be scaled up by adding more video servers.

2) *High availability*—The system can still provide service when some servers fail or are under preventive maintenance.

3) *Competitive performance-to-price ratio*—Personal computers are currently fast and cheap. The distributed server configuration is adopted by a commercial VOD system called *iTV* system.

Firstly, by using distributed multi-server configuration, not only the powerful broadband workstations can be used, but also low bandwidth servers can be utilized in an efficient way in the retrieval process. With some effective retrieval strategies, the clients may retrieve different movie portions from different movie servers, taking into account the bandwidths that servers can afford. Secondly, on a network-based service-rendering environment, by employing the multiple-servers strategy as in [1], the workload can be balanced among the servers [2].

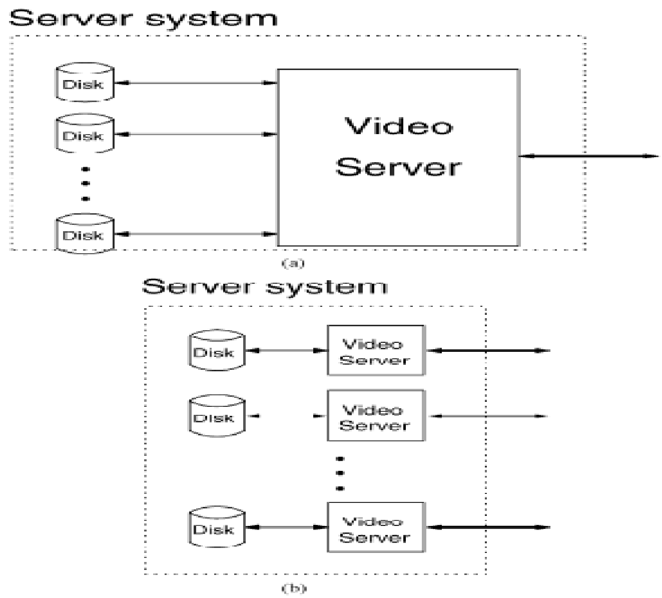

Figure 1.1 Centralized and Distributed Server Configuration

Thirdly, considering the fault-tolerance aspect, even under server/link failures, the workload imbalance can be gracefully taken cared of by the remaining servers. Since multiple servers are engaged in the retrieval process, failure of one or more servers will allow the service to continue without any interruption, so long as there is at least one server active. In contrast, with the conventional system, most probably, the clients may need to be rescheduled, or the presentation gets affected. Also, as shown in the simulation study in [1] , the scalability of the physical system and the heterogeneity of the system, can be easily accounted for in the design, as the size of the portions retrieved from each of the servers depends on the available transmission rate and playback rate of the movie.

1.4 Problem Definition

Our problem domain assumes the following distributed VoD environment. Each server serves its respective local customers and customers situated at other sites. The request for viewing a movie is individually initiated by local clients on each server. Upon the arrival of a request, a server seeks the requested movie locally first. If this movie is available locally, then the movie is retrieved and presented to the user. However, if the requested movie is not available locally, this original server can obtain the information about the requested movie on other servers by employing look-up services, such as the directory service. Then, the requested movie can be retrieved from one or more servers employing our proposed strategy. Below, we briefly describe the basic retrieval mechanism employed in our strategy.

Let a request for a movie is placed locally at the original server (proxy server/scheduler), denoted as, S and let the requested movie, of size L bits, be present at servers S_0, $S_1 \ldots S_n$. Let the connection bandwidths of channels from other servers be denoted as bw$i$, $i = 0, 1, 2 \ldots n$, measured in bits per second, and let the playback rate at the client site be Rp, measured in bits per second. From each server, a portion of the entire movie, denoted as m_i, $i = 0, 1, 2 \ldots n$ is retrieved and is collected by S in a particular order.

Upon receiving the first portion of the movie from S_0, the playback may start at the user terminal, as retrievals from other servers are underway. Presentation continuity is one of the Quality of Service (QoS) requirements for a multimedia presentation. Thus, in order to start the playback when retrievals from other servers are underway, the size of the portion retrieved must be such that there should not be any data starvation for playback. In other words, the size of the portion retrieved must guarantee the presentation continuity.

Now, the retrieval strategy must be such that before the playback of this portion comes to an end, the next portion of the requested movie data should be made available from S_1. This retrieval process continues until the entire movie is retrieved from the set of servers. Along with presentation continuity, our basic objective is to minimize the access time (waiting time for the client) which is defined as the time between a video request is placed to local server to the instant at which the presentation starts at client.

1.5 Thesis Layout

Chapter 2 discusses about VoD architecture, VoD service classifications and its working. In chapter 3 we have surveyed the various QoS parameters needed for distributed multimedia applications, QoS management schemes and QoS for VoD. In chapter 4, we have proposed a retrieval strategy for a distributed VoD system which aims at minimizing the access time of video blocks by keeping the presentation continuity at the client side. Finally, chapter 5 concludes with future enhancements.

Chapter 2
Video on Demand

Video On Demand Chapter 2

2.1 Introduction

With the explosion of Internet, people have endless hype, opinions, forecasts, and beliefs about it. Interactive Television, they feel, is the vision to their beliefs: people will soon be able to purchase products, view movies, play video games, browse Internet, and participate video-conferences without leaving their houses. Of all the new things that people can do with television, video-on-demands is highly supported by Hollywood since it can lead to new markets and can bring them unpredictable profits.

People have been passive participants in receiving what TV service providers offer since television was introduced. Video On Demand (VOD), unlike traditional television delivery, provides users with flexibility in choosing the kinds of information they like to receive . An VOD system is capable of serving a large number of end users to concurrently access large number of repositories of stored data, often movies. In addition to the freedom of choosing movies, users can interact with movies and decide the viewing schedule. In other words, VOD system supports VCR-like functions, such as fast forward, rewind, pause [28]. The enormous communication bandwidth and disk bandwidth required, and the Quality of Service (QoS) demanded necessitate a careful design of the system in order to maximize the number of concurrent users while minimizing the cost.

The 1990's have witnessed an increase in the number of Personal Computers (PCs) in homes and offices, and an increase in the performance of the PCs accompanied by reduction in their prices. This has led to a tremendous increase in the number of computers which have become a part of one global network called the Internet. The Internet, then, made it possible to create a wide spectrum of VoD applications. Some multimedia applications categorized under video on demand are: Movies on Demand,Interactive Video Games,Interactive News etc (Table 2.1).

This variety in VoD services causes one to speculate that VoD systems will become one of the most important services supported by the next generation of computer networks, video servers and distributed multimedia systems [28] . It is also expected that movies on demand applications will soon replace the traditional video rental stores .

Application	Description
Movies on Demand	Users are allowed to select a movie and decide when to play it and invoke VCR-like functions at will.
Interactive Video Games	Users can download games to their machines without the need for purchasing them.
Interactive News	Users can view news from selected top stories
Interactive Advertising	Users can check and purchase commercial
Distance Learning	Students can register classes and view classes
Catalogues Browsing	Users checks the latest available products
Video Conferencing	Users can negotiate with each other
Tele-Shopping	Users can choose from a list of goods and order their item

Table 2.1 VoD Applications

In a report called Present and Future by In-Stat/MDR [26] expectations indicate that by the end of 2004, the number of consumers using family oriented on demand IP services will out number the users of general content services. Also, in the report, the author speculates that by the end of 2009, 40% of worldwide consumers who have high-speed Internet connections to their residences will be using on demand services for which they pay monthly fees. Table 2.2 shows the participation and the share of the countries worldwide in providing VoD services. It indicates that, although Europe is occupying the middle of the world, it needs to spend more in this area of services so that it can compete with other countries

Countries	%
United States	43.3
Asia, especially South Korea, Taiwan,	37.0
Europe	15.0
Rest-of-the-world	4.7

Table 2.2: Percentage of VoD participants

These expectations have encouraged researchers world-wide to investigate and spend in this area with the hope of providing different varieties of VoD applications. They have concentrated on the main issues that affect the performance of VoD systems in one way or another. Such issues include; video on demand services and their characteristics, VoD servers architectures and their internal design, video storage management, video data transmission techniques, and scheduling policies which are responsible for admission control and video data delivery. In addition, the client design and architecture have received great consideration since the video will be played at the client‘s machine were the VoD service will be evaluated.

2.2. System Architecture

A VOD system comprises of 3 major components [28]: the "set-top box" at the client's site, the distribution network, and the server. There are many design issues to consider in building each of these components As with other networked systems, VOD can be designed as centralized multimedia systems or distributed multimedia systems. A centralized VOD system places processing servers and media archives in a single site as a central node. Requests from clients are processed at the central node, and videos demanded are delivered through the network to the client sites. Figure 2.1 illustrates centralized system architecture. Centralized VOD systems are simple to manage, but they usually suffer from poor scalability, long network delay, and low throughput. The performance of centralized VOD systems can be improved if local servers are added. These local servers have video buffers, but no media archives. Popular movies can be stored in local video buffers so that they can be delivered to clients more quickly.

Videos that are not buffered at local sites can be delivered to clients from the central archive when they are requested. A distributed VOD system has local processing servers and media archives. Clients' requests are handled by local servers (figure 2.2). If the movie requested is not in the local archive, the local server can request the movie from remote servers located across the network.

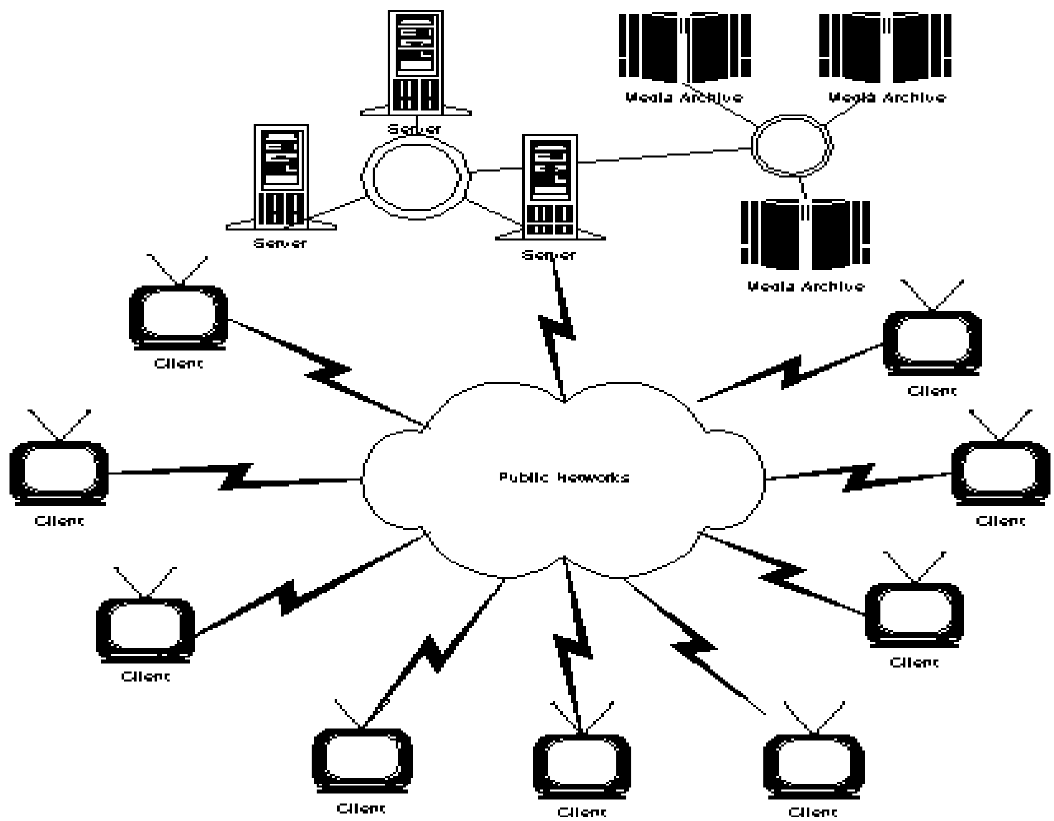

Figure 2.1. A Centralized Video On Demand System

A distributed VOD system can be viewed as many small regional VOD systems connected together. The distributed VOD system spreads users' requests to many sites, thereby moving the processing servers and media archives closer to the clients. Local servers reduce network delay and congestion as experienced by central servers, but distributed systems are more difficult to manage. The choice of the system structure depends on the available storage, communication systems, costs, application demands, and other factors. However, the desired QoS of VOD systems makes the distributed structure more preferable

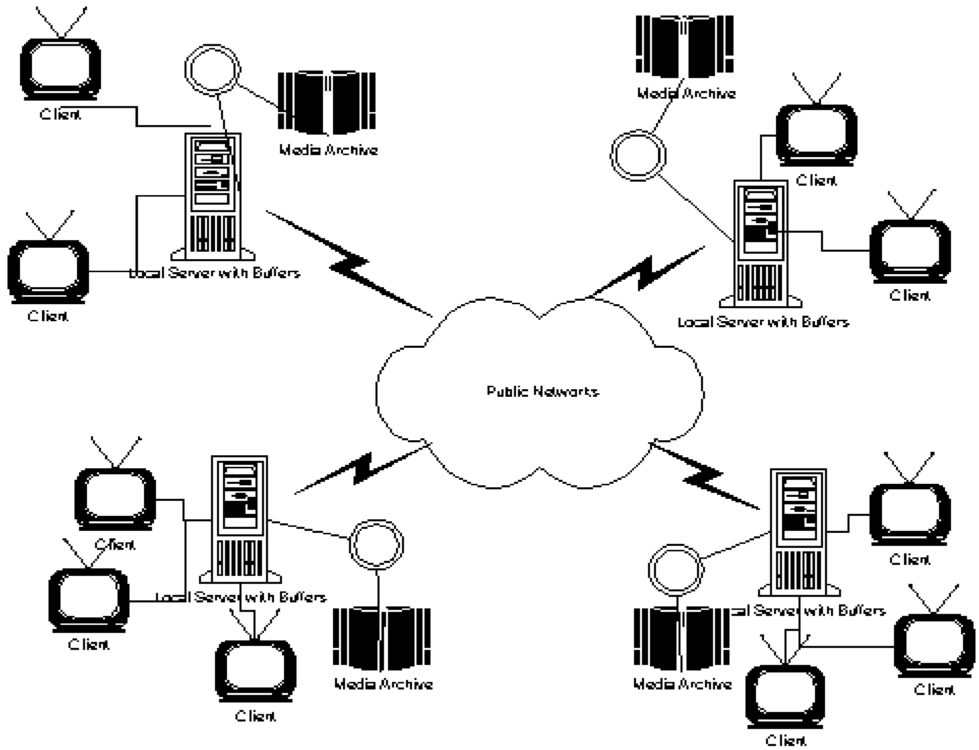

Figure 2.2. A Distributed Video On Demand System

Each VOD connection requires a bi-directional communication between the client and the local server. Each server has a number of video selections available for users. The server processes the client's requests and tries to respond to the clients' demands as soon as possible. A VOD system should be able to handle hundreds or even thousands of clients with different preferences simultaneously [26]. The quality of each service should remain in specific bounds throughout the entire session. A VOD service usually starts from a client requesting information from a server; the server responds via the network to the client.

The system architecture of a Video On Demand system basically consists of three major parts[28]: a client, a network, and a server. Each part can be subdivided further into components and interfaces. Figure 2.3 depicts the communications between clients and servers.

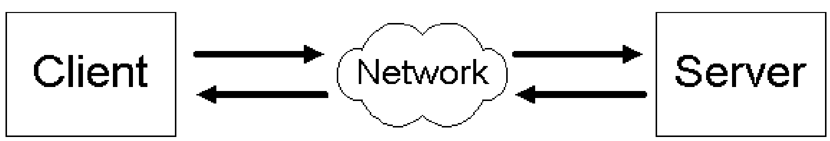

Figure 2.3. Communications Between Clients and Servers

2.2.1 Clients

A client subscribing to an Interactive Video On Demand service has a display device (usually TV) and some audio devices (e.g. speakers) to present the movie requested. He/She interacts with the system via an input device such as a remote control, a mouse, or a keyboard. A controller is needed at the client site to take the client's commands and to send the signal to the server through its network interface. The controller also stores video signals it receives from the server into its buffers, decodes the compressed signals, and sends the decoded signals to the display at the appropriate time. The controller is assembled in a box, known as the "set-top box." Figure 2.4 depicts the components at the client site.

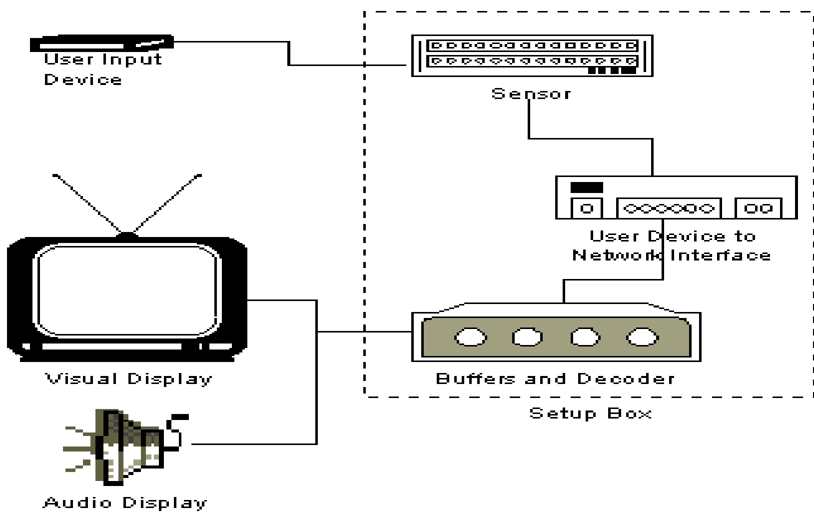

Figure 2.4. A User's Set-Top Box

2.2.2 The Network

A VOD service requires real-time display of the video purchased by the client. A typical video stream consists of frames of pictures, sounds corresponding to those frames, and captioned text. The large quantity of information needed to be transmitted to the client continuously with minimal delay poses high performance requirements on the network. A VOD network should be a high speed network with reasonable error rate as retransmission is unacceptable. Since video information is delay sensitive, the delay variation (jitter) should be kept to a minimum.

2.2.3 The Server

A server of a VOD system processes commands from users. It accepts or rejects the clients' requests based on the current state of the system and the network load. It also performs scheduling on the retrieval of data for all active clients. A multimedia archive is connected to the server. The archive contains a collection of videos available to the users. Depending on the system requirements and the budget available, a range of storage devices can be used: cache (RAM) is the most expensive but has the lowest access time. Disk-arrays provide fault-tolerance at a reasonable price and access rate (10 msec). Optical discs have a capacity of 650 MB with access time 100 msec. Digital Versatile Disc (DVD) is state of the art. Each disc can stores 4.7 gigabytes of information. The content of movies stored on DVD discs can be easily configured to suit viewers' preferences with the help of authoring tools. Tape drives are on the lower price range, but with longer access time. A typical VOD storage system uses a combination of storage devices to optimize the tradeoff between cost and efficiency. Figure 2.5 depicts a general VOD storage hierarchy.

Combining all the components above, a VOD system is constructed. The overall system architecture of a VOD system is shown in Figure2.6.

2.3 Video on Demand Characteristics

The design of such multimedia services differs significantly from traditional text/numeric retrieval services since the playback of digital video and audio consumes data at a very high rate [24][25]. Also, video data needs to be played back continuously in a timely manner in order to preserve its meaning, while text data does not have temporal constraint. Therefore, when designing VoD systems, one should take into consideration the VoD service characteristics such as Long-Lived Session , High Bandwidth Requirements, Support for VCR-Like Interactivity , and Quality of Service[26].

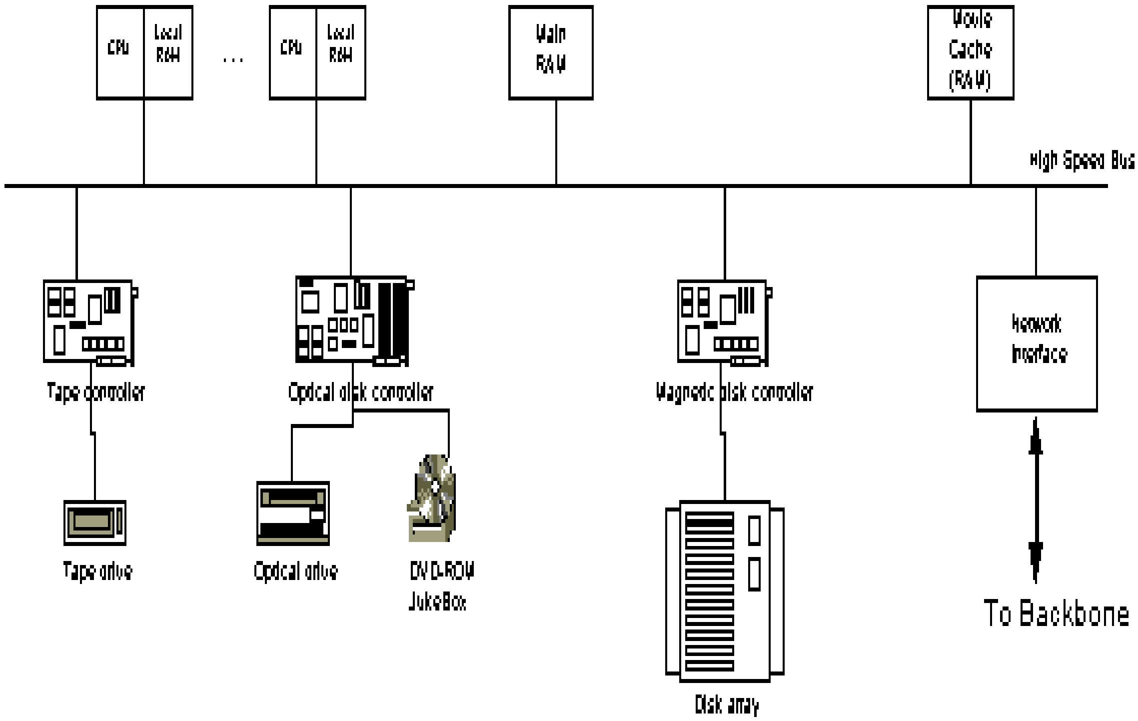

Figure 2.5. A VOD Storage Hierarchy

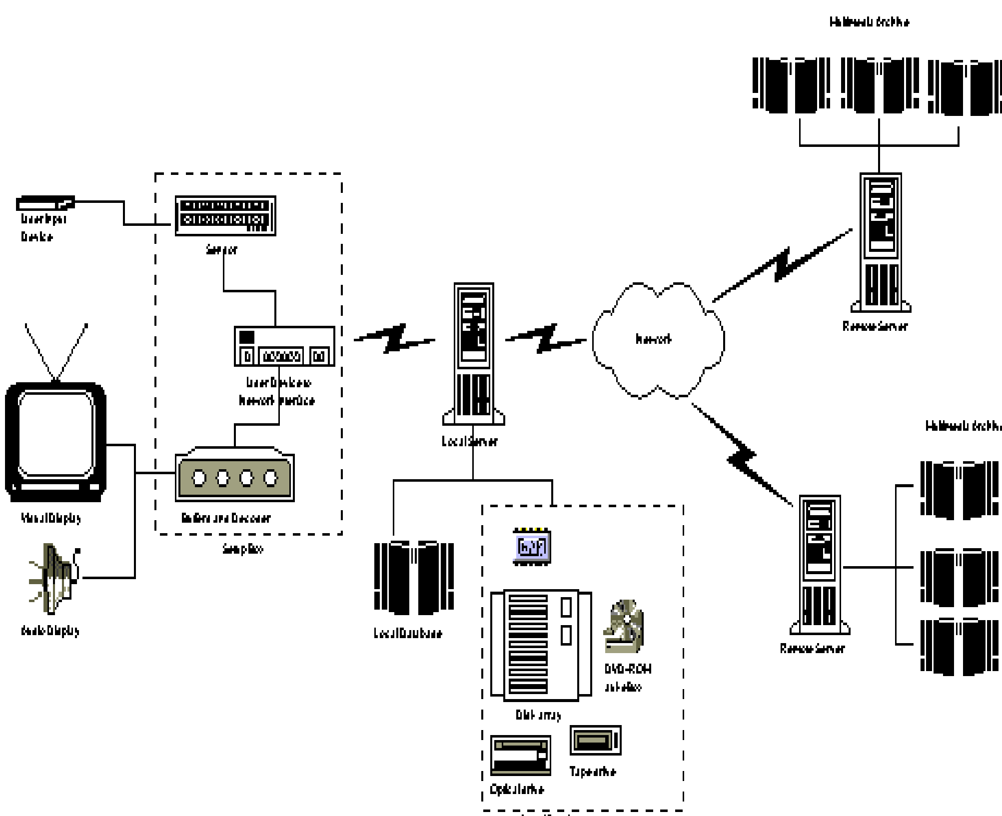

Figure 2.6. The Overall Video On Demand System Architecture

(i) Long-Lived Session:

A VoD system should take into consideration the support of long-lived sessions during the transmission from the server to the client‘s playback machine. This is required since some video files require long time of playback. For example, a typical movie-on-demand service might last 90-120 minutes in servicing a single movie. That is, if the video data is sent at a rate equal to real-time play back rate then the server‘s resources which are needed to maintain this session must be reserved for 90-120 minutes.

(i) High Bandwidth Requirements:

Video data transmission requires high I/O and network bandwidth. For example, the server storage I/O and network bandwidth requirements for a MPEG-1 stream are 1.5 Mbps (mega bits per second). Another example is the MPEG-2 standard which specifies a higher compressed bit streams for high-quality digital video at a rate of 2-80 Mbps. Thus, a two-hour MPEG-2 video with a resolution 720x486 and bit rate 4.5 Mbps requires about 4 GB of storage. Likewise, MPEG-4, NTSC, and HDTV formats also require high bandwidth. Sending such video data from the server to the client without interruption is a great challenge.

(ii) Support for VCR-Like Interactivity:

A client expects the VoD system to offer VCR-like functions, such as the ability to play, forward, reverse and pause. This requirement will increase the cost of the service since it implicates that each user might be assigned a dedicated session in order to give him the total freedom to invoke the VCR functions at will.

(iii) Quality of Services (QoS):

The QoS that VoD consumers and service providers might be concerned about, includes service latency, defection rate, interactivity; playback effects of videos…etc. Therefore, some mechanisms must be implemented to achieve the best QoS. The acceptance of a new user must not affect the QoS provided to the users being serviced, and adequate resources must be available throughout the entire system from the server to the user during the playtime [23][26]. These are general characteristics which play an important role in the design of video on demand services. However, the kind of application designed might relax some of these characteristics. For example, running video clips does not require as long a time session as running typical movies. Also, in some VoD applications, e.g. in advertisement, the interactivity might not be a requirement and therefore does not need to be applied. VoD services enable clients to select a video program, often a movie like you would get from the video rental store, and have it sent to them, in a form called a stream, over a channel via a network such as a cable or satellite TV network. This way, the clients are imitating television viewers. However, the VoD clients can go beyond the typical television viewers since they will be able to interact with the service and invoke VCR-like functions such as; pause, jump forward, jump back, stop, and so forth like they would if it was running on their own VCR or DVD devices. Different from the VoD services, television viewers are passive and can't interact with the service provider and have no control over what they watch, since the service provider is the one who chooses what to broadcast.

2.4 Classification of VOD services

Video on demand application, basing on the level of interactivity of the user, can be classified into following categories[26][28]:

- Live Broadcast (No-VoD) services similar to broadcast TV but is transported over a network as the Internet. The user is a passive participant and has no control over the session.
- Pay-per-view (PPV) services in which the user signs up and pays for a specific program that starts at different preset times.
- Quasi Video-on-Demand (Q-VoD) services, in which users are grouped based on a threshold of interest. Users can perform at the simplest level temporal control activities by switching to a different group.
- Near video-on-demand (N-VoD) services in which functions like forward and reverse are simulated by transitions in discrete time intervals (on the order of 5 minutes). This capability can be provided by multiple channels with the same programming skewed in time.
- True Video-on-Demand (T-VoD) services, in which the user has complete control over the session presentation. The user has full-function VCR (virtual VCR) capabilities,

including forward and reverse play, freeze, and random positioning. This service has one unique streaming to every client that is granted access to the media which means that several customers can start watching the media whenever wanted. To be able to compete with the VCR system it is essential that the advantages of this kind of Video-on-Demand are provided.

From the business point of view, the VoD services delivery tends to be provided in three forms [25]:
Free VoD services where the users can download free video clips such as movies clips, advertisement, and news Subscription VoD services, where users are obliged to pay a monthly or annual fee so that they can access the service (e.g. Showtime On Demand and HBO On Demand) . Pay-Per-View VoD services, in these services the movies are ordered at will and, the users don‘t have to make monthly subscription and pay only for what they watch.

Most likely, the customer would be interested in free delivery of movies on demand service. This service fits perfectly in assumable none profitable organizations and educational institutions such as universities and schools. The users can use an IP address to connect to the video server and browse a list of movies from which they can select their favorite one. From the server point of view, the video file can be downloaded in its entirety and stored at the client‘s side before it can be played. This action would have the following benefits:

Allow the end-user to watch the whole movie from their hard drive. Thus, once the movie is downloaded the user can decide when to play it. Allow the user to invoke the VCR-Like functions at will. This would give him the ability to pause, do other activities and resume. Eliminate exchanging messages produced between the client and the user, as a result of invoking interactive functions by the user. Eliminate the user trips to the video rental store. However, downloading the whole video file and storing it on the client‘s hard drive before it can be played back would definitely have some drawbacks on the server side as well as the client side. These drawbacks are stated as follow:

• The client must have a huge hard drive since the compression bit rate of multimedia data is high (Table 2.3). In the past, such solution was not acceptable since the storage was costly. Nowadays, this can be applicable to some home PCs, but not all the clients have this capability. For example a two-hour MPEG-2 movie with a bit rate of 4.5 MB/s would require 32.4 Giga Byte (GB).

Media type (specifications)	Bit Rate
Voice quality audio (1 channel, 8 bit samples at	64 Kb/s
MPEG encoded audio (equivalent to	384 Kb/s
CD quality audio (2 channels, 16 bit samples at 44.1 kHz)	1.4 Mb/s
MPEG-1 encoded video	512 Kb/s or 1.5 Mb/s
MPEG-2 encoded video	1.5or 5 Mb/s
MPEG-4 encoded video	40 Kb/s or 1 Mb/s
NTSC quality video	27 MB/s
HDTV quality video (1280 X 720 pixels/frame, 24 bits/pixel)	81 MB/s

Table 2.3.Examples of media compression formats

•The response time will be high. That is, the client must wait long before he can watch the movie. For example, downloading a two-hour MPEG-2 movie with a bit rate 4.5 MB/s over 10Mb/s network would require 432 minutes. That is, the user must wait 7.2 hours before he can watch the movie on his playing screen The server performance in term of the number of users connected to it is degraded since most of the user would turn their back and cancel their request if they don‘t get the service in an acceptable short response time. An alternative way to downloading and storing the whole video file at the client buffer is to stream the video data to the client and start the playback right after the arrival of the video data to the player buffer. This would be done by delivering the video data over a network in real time to the user‘s set-top box or PC, and store part of the video in the client‘s machine. At the same time, the server must be able to discover that it has the capability to stream the video data at a rate greater than the real time rate and find out the clients who can benefit from the available resources and send them video data at a rate greater than the video rate consumption. Also, the server should be able to notice that there are new arrivals who request the same movie file at the same time or within a short interval of time between them. In this case, the server must be able to serve all of them as if

they were one request. This way, the number of clients served by the server is increased.

2.5 Working of VOD system

The complete video on demand system can be modeled as shown in Figure 2.7..The major activities involved in a video on demand system are[28]: (i) admission control and resource reservation (e.g. bandwidth allocation), (ii) retrieval of video blocks and (iii) streaming of the blocks to the client.

These basic activities give the VoD researchers an idea about the major issues that need to be handled during the VoD design. Some of these issues are related to the server design and others issues are related to the client design. The network, of course, is also an issue that needs to be taken into consideration.

At the server side, the VoD server should have a mechanism for the acquisition of the request and admission control. The admission control must be applied to all resources of the system. These resources include the CPU, Memory, Disk and Network. If the system has enough resources then the server should provide the service. In addition, the server must take into consideration the file system and the way the video data is stored in the system. Also, the server must define an algorithm for scheduling the requests and a mechanism for the transmission of the video data from the server to the client.

On the other hand, the client must have a mechanism such as a GUI from which he will be able to connect to the server, to browse the video files available in the server, to be able to synchronize with the server based on the transmission mechanism, and finally to display the video data. The following subsections give details of the above mentioned basic activities.

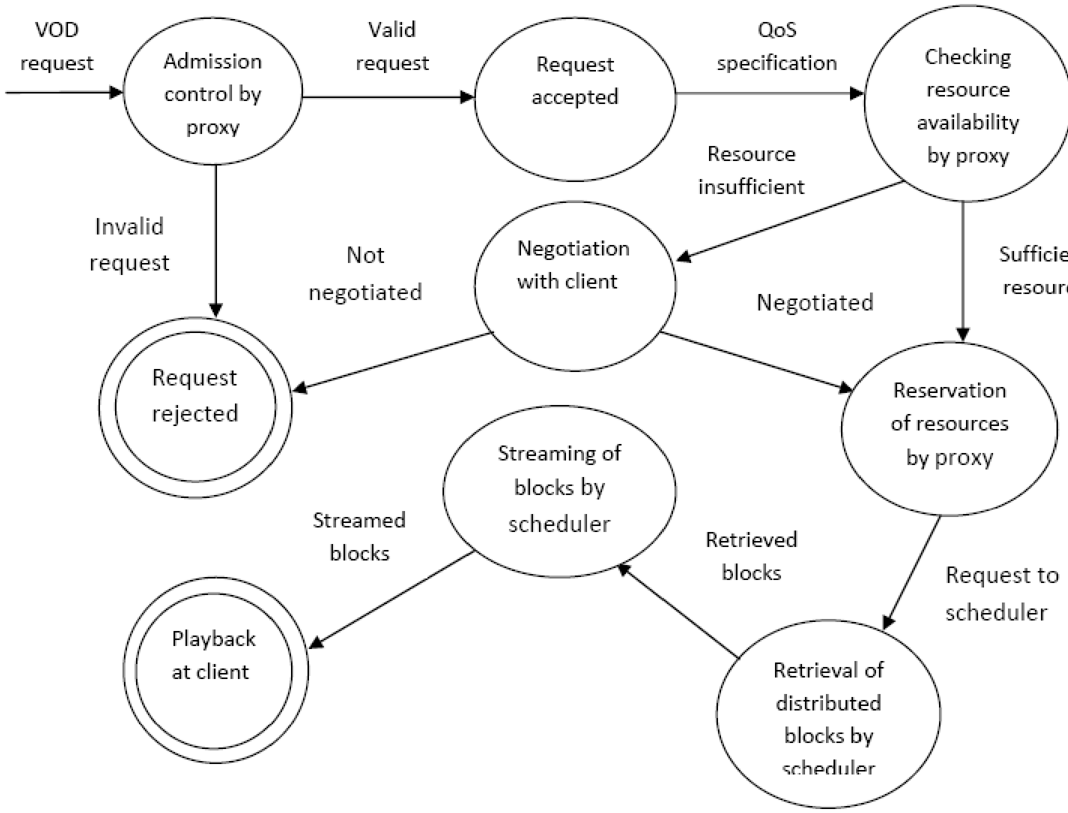

Figure 2.7 VoD working Model

2.5.1 Admission Control (AC)

Given the fact that the video data must be continuous, the server must employ an admission control algorithm before accepting new requests. It must ensure first, that adequate resources are available to the new request throughout the entire path from the video server to the client‘s presentation device, and second, that the acceptance of a new request does not affect the performance requirements of other clients being already in service Also, the admission control should be able to negotiate and renegotiate the client‘s requirements which will be translated into system resources and quality of service. The client is always looking for the best quality which implicates that the server must meet the performance requirements of the clients. Prior to admitting a new client for VoD services, the VoD server must have sufficient resources in order to guarantee that the QoS contracted to existing clients is not jeopardized. To implement admission control, four scenarios can be assumed: Deterministic Server, Statistical Server, Predictive Server , and Background (or Best Effort) Server. These scenarios are explained in more details as follows:

• Deterministic : All deadlines are guaranteed to be met. For this level of service the admission control algorithm considers worst-case scenario in admitting new clients [26]. To implement deterministic service, resources are reserved in worst case manner for each stream admitted.This scenario is used when the client can't tolerate any deadline violation.

• Statistical : deadlines are guaranteed to be met with a certain probability [8][11]. For example, a client may subscribe to a service which guarantees that, 90% of deadlines will be met over an interval. To provide such guarantees, admission control algorithms must consider statistical behavior of the system while admitting new clients. Implementing statistical service would proceed as with deterministic service, but instead of using worst case values in computing the change to round length, some statistical distributions would be used. For instance, instead of using a worst case rotational delay value, an average value may be used, which can be expected some percent of the time based on a random distribution of rotational delays. Providing statistical service guarantees is essential not only due to the variation in the seek time and rotational latency, but also due to the variation in the data transfer requirements of compressed media streams. To provide statistical service guarantees, a server could employ precise traffic characterizations, rather than the worst-case or the average-case values. It is also possible that when variable rate data is stored, a complete and accurate description of the rate change could be computed, so that the server could use the information during playback to reserve only the required amount of the server resources.

• Predictive : The consumed resources rate is predicted from the history, assuming that the past behavior is an indicator for the future [25]. Thus, the server can predict the resources that will be needed in the future and make the scheduling based on this prediction. Although prediction might give high resource utilization, it provides weak guarantees.

•Background(or Best Effort) : no guarantees are given for meeting deadlines. The server schedules such access only when there is time left over after servicing all guaranteed and statistical clients [23].

2.5.2 Video Data Retrieval

The retrieval of video data in multimedia servers can be done based on one of two main paradigms[7][8][9][11]: server-initiated or client-initiated. In server-initiated paradigm, the server generates video streams in a periodical manner with which, the clients will be able to join these streams. In the client-initiated paradigm, the streams are initiated based on the clients‘requests arrival. Regardless of the data retrieval paradigm, the storage server must employ admission control in order to ensure that the acceptance of a new request for data retrieval does not affect the real-time requirements or the quality of service of the streams already being serviced. From the above discussion, it can be realized that great attention must be paid to the way the video data must be stored and distributed among the server storage, the way the video data is retrieved, and the amount of the video data that should be retrieved in a timely manner. These factors make multimedia storage servers different from conventional storage servers. These differences come from the fact that multimedia systems require different scheduling algorithms and paradigms to be adopted. Even the choice of the disk hardware affects the storage server since they influence the storage server throughput and play a great role in the success of providing video on demand services.

2.5.3 Video blocks Streaming

As mentioned before, a key issue in VoD service is the ability to stream continuous video data from servers to clients across the network providing TVoD service without delay and playback interruption (jitter). Fifteen years ago, most multimedia pundits were predicting that VoD would radically change our home entertainment habits. However, none of the companies that invested in VoD have been able to come up with a single successful commercial system since VoD is still too costly to compete with either video rental or pay-per-view television. In 1998/1999 Time Warner Cable Comcast piloted a service using Scientific- Atlanta‘s (SFA) —Explorer 2000“ digital set-top box and a SUN MicroSparc/Power TV based STB. Trials by Bell Atlantic and Time Warner proved that though streaming video-on demand was possible, the capital expense of 7000 pound per video stream (per end-user) did not justify the business model [25]. If a provider wanted to sell each stream at 2 pound (3/) (i.e. 3/ per movie), viewers would have to buy two movies every day for five years just to cover the initial expense. However, the price for video-on-demand servers has been subjected to Moore‘s Law, and what used to cost 10500 / now costs less than 1050 [7]. The VoD service is still costly due to the high requirements of server and bandwidth resources (in particular, server I/O bandwidth and network bandwidth). The long-lived nature of digital video would hold these channels for long a time. For example, if a typical two-hour movie is played back at a rate equal to real-time playback rate, then the video channel will be occupied for two hours, and the server must maintain streaming the video data transmission for two hours. This means that the server's resources are also occupied for two hours. The problem is further complicated by the fact that conventional video streaming systems use a linear playback scheme that forces users to download from the beginning of video. Also, during the playback, the clients may wish to invoke one of the VCR-like functions such as pause, stop, jump forward and jump backward, and they expect that the server should be able to respond

without delay. These factors might have been the reason behind limiting the widespread usage of video streaming over the Internet. Therefore, in order to reduce the VoD service cost and to provide interactive video-on-demand, the server must employ appropriate techniques to efficiently utilize the server resources and stream media data to the clients. In general, there are two approaches for allocating server channels for the delivery video data called: user-centered

approach and data-centered approach

User-Centered Approach : A conventional VoD system assumes the user-centered scheduling scheme in which a user eventually acquires some dedicated bandwidth [8] [9]. The consumption rate of a video object is equal to the amount of bandwidth necessary to view it continuously. When a client makes a request to the server, the server sends the requested object to the client via a dedicated channel. This scheme incurs high system costs, especially in terms of server storage-I/O and network bandwidths. However, some techniques can be implemented to reduce the cost .

Data-Centered Approach : Data-centered scheduling dedicates channels to video objects, instead of users. It allows users to share a server stream by batching and using the multicast facility of modern communication networks. Also, it has the potential to dramatically reduce the network and server bandwidth requirements. The data- centered multicast VoD service can be either client-initiated or server-initiated [11] [12].

• In the client-initiated service, channels are allocated among the users and the service is initiated by clients so, it is also known as a scheduled or client-pull service.

• In the server-initiated service, the server channels are dedicated to individual video objects, so it is also called a periodic broadcast or server-push service. Popular videos are broadcast periodically in this scheme, and a new request dynamically joins, with a small delay, the stream that is being broadcast.

• Another option would be combining the above two schemes. This combination is called hybrid batching. In practice, it is efficient to use hybrid batching that combines the above two schemes.

2.6 Conclusion

A VoD system is usually conceived as a system having a single video server. However, with an increase in the user access rates, the network bandwidth becomes a natural bottleneck, especially for a network based multimedia service. A decentralized (distributed) approach would rather handle this increased access rates more elegantly. Further the design of such multimedia service differs significantly from traditional text/numeric retrieval services since the playback of digital video and audio consumes data at a very high rate. Also, video data needs to be played back continuously in a timely manner in order to preserve its meaning, while text data does not have temporal constraint. Therefore, when designing VoD systems, one should take into consideration the VoD service characteristics such as long-lived session, high bandwidth requirements, support for VCR-like interactivity, and Quality of Service.

Chapter 3
QoS in Multimedia

3.1 Introduction

It is a fundamental truth that the quality and quantity of any data delivered to a user is limited by the quality and quantity which the underling data transfer system(s) can support. It is also true that, in general, there is a mismatch between the theoretical maxima and those which are available in practice, due to factors such as the sharing of resources with other users, the overheads generated by the various communications tasks etc. The implications of this from a user's perspective are that any applications which rely on the transfer of data are limited by the ability of the data transfer systems in terms of speed, reliability and accuracy. This places a requirement on the developers of these applications to have an awareness of the impact of these limits, and to design their systems accordingly. However, the situation is made more complex by the fact that the quality of service (QoS) available varies from network to network, and may also vary over time on the same system, due to the need to share resources between a variable number of other users. It is therefore important to be aware of the processes by which QoS can be determined, negotiated and varied before or during the operation of an application.

3.2 Overview of QoS

3.2.1 Definition

The ISO/IEC recommendation X.901-5, Open Distributed Processing Reference Model (RMODP) provides a de-jure definition of QoS. This subsection presents an overview of the most critical parts of what QoS is, and what it entails, based on [15][16][17] whose treatment is primarily concerned with multimedia .Traditionally, the term Quality of Service has been used to designate certain technical parameters in the specification of communication protocols. According to a more general definition, found in the Reference Model for Open Distributed Processing [15] QoS is ‘‘A set of quality requirements on the collective behavior of one or more objects.’’ However, this type of definition seems to be too general to be meaningful for our purposes, since it tends to include all system parameters without distinction. We will use the following working definition: By Quality of Service we mean the set of those technical and other parameters of a distributed multimedia system, which influence the presentation of multimedia data to the user, and in general the user's general satisfaction with the application. Based on this definition, the following sub-section differentiates and characterizes the various QoS parameters.

3.2.2 QoS Characteristics

While systems are often defined in terms of their functionality, QoS defines non-functional characteristics of a system, affecting the perceived quality of the results. In multimedia this might include picture quality, or speed of response, as opposed to the fact that a picture was produced or a response to stimuli occurred. Table 3.1 shows the main technology-based QoS parameters which we surveyed, and Table 3.2 summarizes the main user-based parameters. [18] describes perceived quality as user level QoS requirements, and then maps them to lower level QoS characteristics. [19] describes a selection of quality characterizations in terms of QoS parameters and value ranges, for various data types.

Cost is a slightly different category to the others described, as it is not an intrinsic part of the visible results of most transactions. Cost is generally described in terms of a monetary value per interaction, or in terms of the time spent interacting. It is often the case that cost will be used to place upper and lower limits on other characteristics. For example, I am prepared to pay Rs30 per hour to watch this film, only if the quality of reproduction is at least half that of terrestrial television, but I will not pay more than Rs50 per hour, however good the quality of the results, and I want a refund if it stops half way through. Note that the quality of reproduction in this example will probably be decomposed to terms such as jitter, frames per second, resolution and color depth. [20] includes treatment of this characteristic, but found that where multiple choices are possible within QoS requirements, the desired trade-off when incorporating cost may not be clear if requirements are not prioritized. [16] discusses pricing policies in relation to users, and observes that describing pricing in terms of application level characteristics is required. [19] discusses negotiation of a specification based on the perceived worth of the various alternatives offered to the user. It the model described the information source is not given a description of the client's worth descriptions for various parameters. However, it is not hard to imagine that one parameter being negotiated over is cost, or that the server could offer alternatives with various costs coupled with variations in other parameters corresponding to levels of perceived QoS from market research. Once user preferences are classified it would not be complex to offer standard packages based on the market's willingness to pay for a level of service, rather than the cost of providing that service, as already happens in many other industries.

Category	**Parameter**	**Description / Example**
Timeliness	Delay	Time taken for a message to be transmitted
	Response time	Round trip time from request transmission to reply receipt
	Jitter	Variation in delay or response time
Bandwidth	System level data rate	Bandwidth required or available, in bits or bytes per second. Basic mathematical models for
	Application level data rate	Bandwidth required or available, in application specific units per second, e.g. video frame rate
	Transaction rate	Operations requested or capable of being processed per second
Reliability	Mean Time To Failure (MTTF)	Normal operation time between failures.
	Mean Time To Repair (MTTR)	Down time from failure to restarting normal operation
	Mean Time Between Failures	MTBF = MTTF + MTTR
	Percentage of time available	MTTF / MTTF + MTTR
	Loss or corruption rate	Proportion of total data which does not arrive as sent, e.g. network error rate

Table 3.1 Technology-Based QoS Characteristics

Category	Parameter	Description / Example
Criticality	Importance rating	Arbitrary scale of importance, may be applied to users, different flows in a multimedia stream, etc.
Perceived QoS	Picture detail	Pixel resolution
	Picture colour accuracy	Maps to colour information per pixel
	Video rate	Maps to frame rate
	Video smoothness	Maps to frame rate jitter
	Audio quality	Audio sampling rate and number of bits
	Video/audio synchronisation	Video and audio stream synchronization, e.g. for lip-sync.
Cost	Per-use cost	Cost to establish a connection, or gain access to a resource
	Per-unit cost	Cost per unit time or per unit of data, e.g. connection time charges and per query charges.
Security	Confidentiality	Prevent access to information, usually by encryption but also requires access control mechanisms
	Integrity	Proof that data sent was not modified in transit, usually by means of an encrypted digest.
	Non-repudiation of sending	Signatures to prove who sent or received data and when this occurred
	Authentication	Proof of identity of user or service provider to prevent masquerading., using public or secret encryption keys.

Table 3.2 User-Based QoS Characteristics

The *Security* requirements, indicated in Table 3.2, may be specified as a QoS requirement in terms of discrete classes or levels as for other QoS parameters. We shall not examine security mechanisms here, as it forms a large and separate topic in itself.

3.3 Class Of Service

A further important classification of QoS requirements, or more particularly the systems implementing the requirements, is the class of service provided. [20] subdivides classes of service (CoS) into five levels:(i) Deterministic guarantee (ii) Statistical guarantee (iii) Target objectives (iv) Best effort (v) No guarantee

These are not the only classes of service in common use. Many networking standards use CoS to describe levels of service for other parameters than the level of guarantee given to QoS specifications, as we are using. For instance, [15] describes the 5 CoS levels in the ISO/OSI reference model (ISO 7498) to describe transport layer protocol classes.

Deterministic guarantees will always be met or bettered, under all circumstances, while a *statistical guarantee* allows a percentage of time where the guarantee is not met. The last three levels provide no real guarantee, but offer varying levels of assistance in achieving the desired QoS. A system which takes account of *target objectives* will try to satisfy requirements, with some knowledge of their implications, which could then be used to determine scheduling priority. A *best effort* system, like the Internet, would provide the same QoS for all services i.e. with no real consideration of QoS factors. Some historic information about performance is then the only guide to the level of service to be expected, although there is a move to provide some QoS guarantees within the Internet. *No guarantee* is a similar class to best effort, although it is unlikely any information about system performance is available with this class of service. There are often limits to the degree of guarantee available, for instance under multiple failure conditions, however both the above guaranteed classes become important under high load and overload conditions. From the examples given here, and in the literature, it is apparent that all parts of a system must perform their jobs to a certain standard, to achieve a given overall QoS. This end-to-end specification of QoS has been a major practical hurdle, as while network technology e.g. ATM has well established QoS defining characteristics, often many parts of the internet do not actually have the capability of supporting QoS specifications. QoS management is defined by Blair as “the necessary supervision and control to ensure that the desired quality of service properties are attained and (where applicable) sustained. QoS management applies both to continuous media interactions and to discrete interactions. It is this management of systems for QoS which is described in the following sub-sections.

3.4 QoS Management

The various aspects of interaction and types of guarantees required, as described above, must then be synthesized into a specification of requirements, and relationships for trade-offs to enable the delivered QoS to be managed. We divide these first into static and dynamic functions: those which are applied at the initiation of an interaction, and those which are applied continuously or as needed during an interaction.

3.4.1 Static QoS Management

The static QoS management functions relating to properties or requirements which remain constant throughout some activity, are summarised in Table 3.3 and expanded below, drawing
from [15][16][19]

QoS specification is the creation of a contract between producers and consumers of data, based on a specification of requirements. As described above, requirements may be at various levels of abstraction from user to low-level descriptions, and describe a range of interrelated characteristics. This may be the starting point for negotiation. [20] discuss some psychological aspects of user level QoS specification, and the variation in user level specifications between groups of users with different levels of technical knowledge, temperaments and in different situations. Each element in an interaction may specify what it requires and what it is capable of delivering for each QoS parameter it manages. [19][21] are interesting examples of work on languages for the specification of QoS requirements, and behaviour in relation to actual QoS experienced.

QoS negotiation is the process of reaching an agreement between parties in an interaction, on the acceptable bounds on the QoS to be delivered, to form an agreed contract between all parties. This function must consider the specifications and dependencies of all parties, and may reject a contract, or submit a different proposal from that requested, if the original set of specifications cannot be honoured on an end-to-end basis. [21] uses a notation of user specified weights on parameters to enable some automatic trade-off to take place, and [18] describes the use of a ‘worth’ based mechanism, which combined with the use of constraint specification and weighted parameters allows a general mechanism for selection of a specification from a set of

Function	Definition	Example Techniques
Specification	The definition of QoS requirements or capabilities.	Requirements at various levels of abstractions are described as combined parameter, value, allowed variation, and guarantee level descriptions.
Negotiation	The process of reaching an agreed specification between all parties.	A comparison of specifications in admission control with modification of requirements on failure, and resource reservation when an agreement is reached. The modification of requirements should consider the interrelation of parameters and preferences of the user.
Admission Control	The comparison of required QoS and capability to meet requirements.	The available resources may be estimated with the aid of resource reservation information, and performance models.
Resource Reservation	The allocation of resources to Connections, streams etc.	A time-sliced model of capacity reserved is common.

Table 3.3 Static QoS Management Functions

alternative possible provisions and a set of requirements. There are many algorithms described in the literature, which cannot all be covered here.

Resource reservation is a complementary function to admission control, where agreed requests are registered, and the required resources allocated from the available pool. This then assists in predicting and guaranteeing the performability of requests by tracking expected system usage. [16][18] all provide a similar and elegant solution for resource reservation and admission control in a distributed environment (Degermark's using RSVP, Ferrari's in the Tenet Real-Time Protocol Suite 2), with particular emphasis on providing for advance reservation, and resource use with specified time limits or duration. The increase in the information available provides a significant improvement in the acceptance rate of requests for bandwidth due to better planning. This protocol also provides for more general immediate, unspecified duration channels alongside and dynamically sharing bandwidth with those with advance reservation, and specified duration. [19] describes the IP based protocols of ST-II and RSVP, which provide basic resource reservation in the context of multicasts.

These four static management functions all depend on being able to specify requirements and the current state of the system in an appropriate manner. This may be achieved with deterministic guarantees, for "hard" specifications, or using probabilistic or stochastic specifications. Deterministic requirements specify a precise value or range of values to be achieved for a given characteristic. The probabilistic and statistical methods require a value or range of values to be met for a given proportion of events or time. This is likely to be a more realistic specification than the deterministic specification, which often requires significant over reservation of resources to achieve.

In determining requirements and agreeing to contracts it is important that the end-to-end nature of the requirements is considered. For instance, a video server may be able to computationally service a frame rate which neither its disk interface nor all parts of the network passing the data to the recipients can sustain. In some situations it is necessary to consider human users as part of an end-to-end system, treating them as active participants, rather than passive receivers of information. End-to-end QoS provision including description in terms of the user's perceptions is required, as it is the user that ultimately defines whether the result has the right quality level.

3.4.2 Dynamic QoS Management

The dynamic aspects of QoS management respond to change within the environment, allowing a contract to be fulfilled on an ongoing basis. Also, as [15] notes specifications are often inexact as resource usage and flow characteristics are not generally completely defined in advance. The dynamic management functions are summarised in Table 3.4 and expanded below, drawing from [15][18][19].

Policing is concerned with ensuring that all parties adhere to their part of the service contract. For instance, where a video frame rate of 25 frames per second is required, the provider must not consistently provide too few frames, or generate bursts which saturate some part of the system .

Renegotiation is something of a last resort response to sustained failure to honour a QoS contract. [18] suggests taking the average QoS provided over an interval to avoid spurious renegotiation due to very transient fluctuations. A tuned probabilistic or stochastic model could incorporate current information about the causes of failure and resource characteristics. [21] describes client controlled renegotiation and examines network utilization. They demonstrate graceful QoS degradation of deterministic guarantees, which they contend shows benefits over statistical guarantees. Renegotiation can be invoked by users, on deciding that they do not consider a given characteristic to be acceptable as specified, having experienced it. Renegotiation may also be invoked on long running tasks, where the characteristics of the underlying system vary with time. system where swapping between versions during presentation, while maintaining position in the record, is required. Various scaling techniques are examined in [16]. In general, tasks such as changing resolution of an MPEG video stream in real time are computationally expensive. [21] describes the use of an algorithm for adapting multimedia flows in a multicast application in response to monitored network conditions. [18] gives an example of a control-theoretical approach to adaptation. Many other algorithms and techniques exist in this field, each being designed with a particular application and/or network scenario in mind. We do not intend to discuss these at great length here, although we believe that the use of application level adaptation is a key feature in QoS support for mobile computing systems. Most of the literature

Function	Definition	Example Techniques
Monitoring	Measuring QoS actually provided.	Monitor actual parameters in relation to specification, usually introspective.
Policing	Ensuring all parties adhere to QoS contract.	Monitor actual parameters in relation to contract, to ensure other parties are satisfying their part.
Maintenance	Modification of parameters by the system to maintain QoS. Applications are not required to modify behaviour.	The use of filters to buffer or smooth streams, in order to maintain stable delay, data rate and jitter [19]. QoS aware routing to maintain network characteristics. Scaling media, e.g. by modifying levels of detail provided within a stream.
Renegotiation	The renegotiation of a contract	Renegotiation of a contract is required when the maintenance functions cannot achieve the parameters specified in the contract, usually as a result of major changes or failures in the system.Usually invoked by exceptions raised by the monitoring, policing and maintenance functions.
Adaptation	The applications adapts to changes in the QoS of the system, possibly after renegotiation.	Application dependent adaptation may be needed after renegotiation or if the QoS management functions fail to maintain the specified QoS. Often achieved by media scaling.

Table 3.4 Dynamic QoS Management Function

discusses maintaining a QoS contract under adverse conditions, or reducting a data stream to stay within limits. However, it should also be noted that QoS functions may be applied to increase data transfer rates when the system improves its ability to provide a service, i.e. a quality ordered

sequence of alternatives due to media scaling or renegotiation may be traversed in the directions of both improvement and degradation when QoS passes given thresholds.

3.5 QoS translation

QoS requirements for applications are typically end-to-end requirements which impose corresponding performance demands on both the network and the end-systems/applications. QoS parameters specify the resource quantity allocated to the service, as well as the service disciplines managing the shared resource. Therefore it is crucial to translate the user/application QoS into network QoS. The translation between user QoS and application QoS is nontrivial and still an open issue, because the perceptual issues are not completely understood [15]. Application QoS should be translated to the network QoS. The user specifies the application QoS. Then the communication system will map requirements into a set of system, protocol and network QoS specifications. This translation process is illustrated in fig 3.1

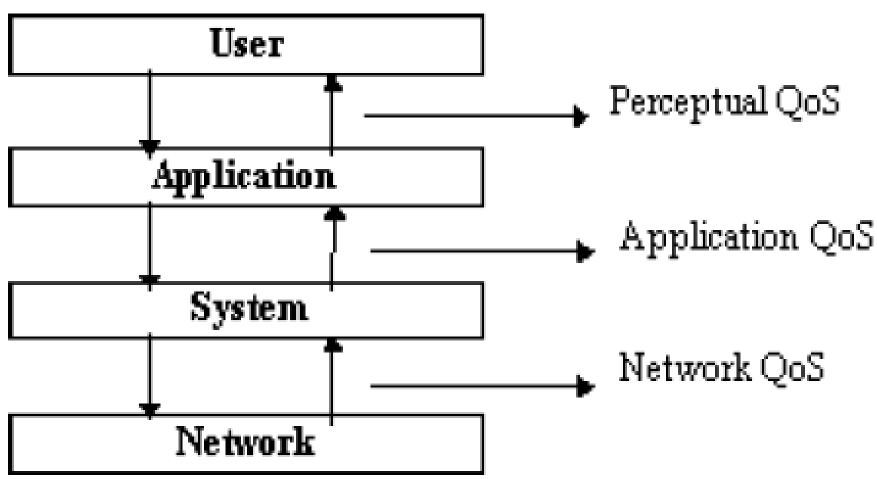

Figure 3.1 QoS Mapping

Fig.3.1 clearly shows that a user or an application specifies requirements; a communication system is responsible for meeting these specifications, and possibly requests an appropriate network resource to the network. The relationship between the application QoS and the network QoS is important because the application QoS can be very different from the network QoS. These QoS differences should be considered in the set-up stage between the user and the network provider. QoS parameters must be mapped to the resource requirements and the required resources must be determined, reserved and allocated along the path between the application and the provider/peer application.

3.6 QoS for VoD

Table 2 summarizes the five categories of QoS parameters [24] required for VoD. The QoS is a difficult issue in that the relationship between application QoS parameters and network QoS parameters is very complex; QoS must be end-to-end; and the application QoS might change during connections. .

Category	**Example Parameters**
Performance-oriented	end-to-end delay and bit rate
Format-oriented	video resolution, frame rate, storage format, and compression scheme
Synchronization-oriented	skew between the beginning of audio and video sequences
Cost-oriented	connection and data transmission charges and copyright fees
User-oriented	subjective image and sound quality

Table 3.5 Categories of QoS parameters for VoD

QoS parameters can also be partitioned into two subsets, namely application-dependent parameters and application-independent parameters. They can also be put into three layers[23]: user, application, and system layers as shown in Table 3.6. The system parameters can be further classified into two categories: network and operating system parameters and device parameters. For multi-media presentation, the quality of audio and video is important in addition to images, text and numbers. Application parameters describe requirements for application services and are specified in terms of media quality and media relations. Media quality includes source/destination characteristics such as media data unit rate, and transmission characteristics such as response time.

Media relations specify relationships among media, such as media conversation, inter-stream synchronization, and intra-stream synchronization.

QoS Layer	QoS Parameters
Application	Frame rate, frame size and resolution, response time ,throughput, security, price and convenience
System	Buffer size, process priority, scheduling policy, caching policy, time quantum
Network	Bandwidth ,throughput, bit error rate, end to end delay, jitter

Table 3.6 Qos layering for VoD

3.7 Conclusion

QoS is one of the major issues for any distributed multimedia application. It is the collective effect of service performances which determine the degree of satisfaction of a user of the service. This implies that the user is the final arbiter of 'good' or 'bad' QoS. Different applications demand different service qualities. VOD is one of those emerging distributed multimedia application in which one of the most challenging aspects is to serve the user in minimum waiting time. The waiting time (response time) is one of the important, application level quality of service (QoS) parameter. If this has to be translated to the system level parameter, then the appropriate parameter will be the access time (retrieval time) of the video blocks which are distributed in different servers. We proposed a novel retrieval strategy which minimizes the access time of the distributed video blocks and hence, provide a feasible waiting time to the user. Another QoS parameter we considered in our proposal is the presentation continuity at the user side.

Chapter 4
The Proposed Retrieval Strategy

Retrieval Strategy Chapter4

4.1 Introduction

In this chapter, we will introduce a novel retrieval strategy that particularly suits long duration media such as feature-length movies (typically longer than 60 and usually around 90 minutes in duration) for Video-on-Demand (VoD) or Movie-on-Demand (MoD) applications'. Below we will see the need for such a multiple server technology for modern day high-bandwidth applications. With the increasing popularity of multimedia services on network based environments, there is a continuous thrust in achieving an optimized design for multimedia servers and network service providers. The main attraction of these services is that viewing and presentation control is handed over to the user, in contrast with conventional video broadcast services as cable TV. A particular movie can not only be made available to a user at his/her convenience (as with video cassettes), but the user can also have complete control on all aspects of the different media involved (audio, video), as for example the physical layout of the screen. Also, with an increase in demand for a particular movie, depending on the popularity profile, viewing cost per user can also be reduced considerably when clever placement of movies at strategic locations on the network is carried out. The effort for the development of such systems and services would be futile without the availability of high performance computers and high speed fiber optic networks that offer the capability of supporting such demands.

The above mentioned application and other (futuristic) applications like MoD, collaborative video editing and synthesis of multimedia objects and other network based distributed applications will be attractive only when the available network bandwidth and other necessary resources are cleverly utilized. Owing to the continuous thrust in developing multimedia services on network based environments, the service providers situated at geographically large distances can co-operate and share their documents in order to serve their local subscribers. Once a document is available locally, in turn, each service provider can choose the appropriate admission control and scheduling algorithms to maximize their servicing capabilities. We believe that this multi-tier service architecture, provides an elegant solution for developing such multimedia services, since the data sizes that are involved are very large.

4.1.1 Service Model

We envision a network consisting of a set of service providers serving each locality (figure 4.1). Each service provider has a directory facility, which registers the available documents at various sites. Whenever a user requests a movie, this directory service will produce a list of servers that can supply the requested multimedia document. Thus, if the requested multimedia document is not locally available, the service provider will request the other service providers to upload that document to its local site. Thereafter, whenever a request for this document arrives, the local server can use the stored document. This service provider could be the multimedia server itself, if it has adequate resources for supporting this service. In such a large network, the user requests may originate anywhere, and servicing these requests should incur the minimum possible delay. If not, such a multimedia service becomes less attractive.

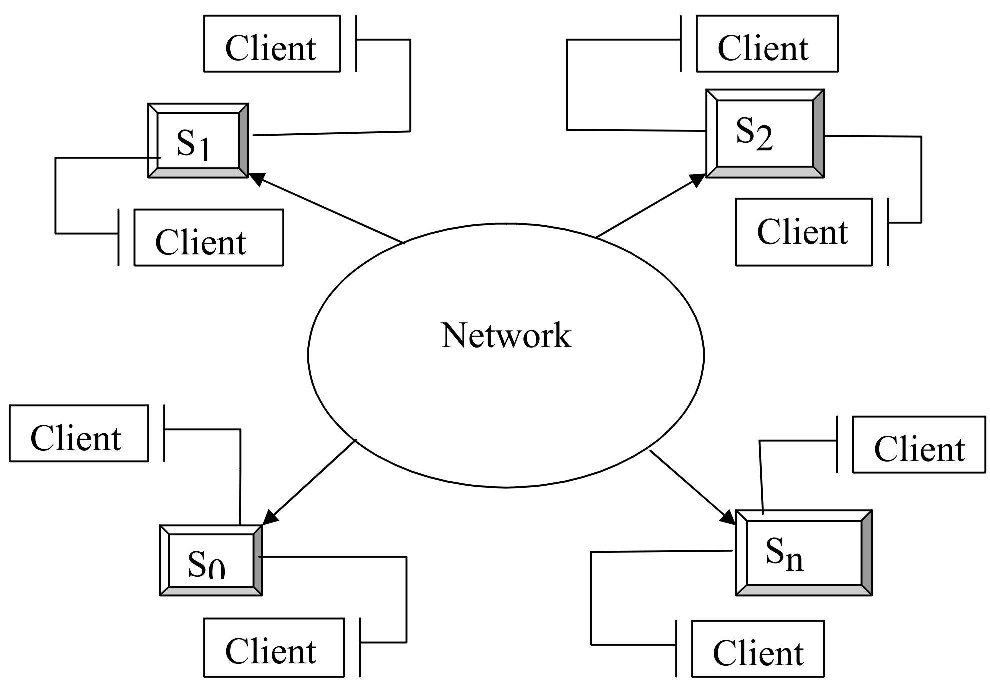

Figure 4.1 Service Model for VoD

In the case of networks that span a large area wherein the server sites are geographically distributed, communication from one site to the other will incur a finite amount of non-zero delay. One of the attractive features of such multimedia services on networks lies in keeping up the promise of a smooth presentation without any audio-visual discontinuities. Unless clever strategies are adopted in retrieving the video blocks amidst the presence of these communication delays, this objective may not be met.

We model the communication delay as a quantity that is directly proportional to the length of the video data that is carried over that established communication path (virtual or circuit switched) to the service provider. This is unlike the model proposed for pyramid broadcasting technique, in which the video blocks in the successive retrievals are of increasing sizes.

In the multiple server technology[1][2][3][5], the available communication bandwidth and the display/playback rate of the video clip are the two major system parameters that are considered, and this chapter focuses in the design and analysis of retrieval strategies that minimize the access or the wait time of the users.

4.1.2 Advantages of Distributed Retrieval Strategy

We now highlight some of the key inherent advantages in using a multiple server approach. Since this strategy primarily involves more than one server for retrieving the document and rendering the VoD/MoD service, this strategy, and hence the technology associated with this service infrastructure is referred to as *Multiple Server Retrieval (MSR)* strategy. A MSR scheme inherently subsumes the following advantages.

Firstly, on a network based service rendering environment, if a single server system, however sophisticated it may be (in terms of speed and capacity) is used there is a continuous "work pressure" that is enforced on the system[4][5]. For instance, when there is a continuous demand for a long duration video retrieval by several clients, a significant amount of the time is spent in servicing these requests, while some small number of requests demanding short services may undergo long waiting times. By employing a MSR strategy, the work pressure can be balanced among the servers [1][2].

Secondly, by using a MSR strategy, even low-bandwidth or heavily-loaded servers, that may not be usable on their own, can now be significantly contributing to a group of several servers that upload a movie [7]. Thirdly, considering fault-tolerance aspects, even under server/link failures, the workload imbalance can be gracefully taken care of by the remaining servers, in a multiple server environment. Since multiple servers are engaged in the retrieval process, failure of one or more servers, will allow the service to continue without any interruption so long as there is at least one server operational [6]. In fact, with a clever design of a retrieval strategy, the clients will continue to view the presentation while a certain number of servers may "die" and come back to "life" after some time. In contrast, with a conventional system, the clients will most probably need to be rescheduled at the expense of their presentation continuity.

Also, as shown in the literature [1], scalability of the physical system and heterogeneity of the system, can be easily accounted in the design, as the size of the portions retrieved from each of the servers depends on the available bandwidth and playback rate of the movie. In effect, a MSR strategy has a natural load balancing capability built-in its design. Each server participates according to its available capacity and/or its connection bandwidth to the client, collectively offering a service far superior than anything it could offer on its own.

Finally, from service provider's perspective, since each server, on the whole, is engaged only for a short while in retrieving a portion of the media document, the number of clients that can be entertained simultaneously can be potentially maximized [7].

4.2 Problem Specification

In this section, we present the problem more formally, describe the network architecture that is considered, and introduce the necessary definitions, notations and terminology. The network model consists of a pool of TV multimedia servers each serving their respective customers (figure 4.1). The requests for viewing a movie of a long duration (typically of 100 to 120 minutes) arrive at these servers from its local customers. These servers are by and large, powerful workstations with sufficient amount of bandwidth capacity and memory space to serve a maximum number of users concurrently by employing efficient admission control algorithms.

Upon an arrival of a request, the server seeks the requested multimedia document. We interchangeably use the terms service providers and servers as per the context. If the document is available locally, then usual retrieval and presentation techniques as described in the so far literature can be employed to serve the request. However, if the requested document is not available, then the server with its directory service facility, a kind of look-up table procedure, determines the server sites at which the requested multimedia document is present. It then obtains a set of server addresses from which the document may be retrieved. The requested multimedia document is then retrieved by employing a MSR strategy demonstrated through the following motivating example. We introduce the necessary notations and terminology in the example for the ease of understanding.

4.2.1 Motivating Example

Consider a scenario in which a requested multimedia document is not available locally at a server denoted as, *S.* Let the requested multimedia document be present at the sites S_0, S_1, and S_2. Let the total size of the movie requested be L = 1*GB.* Further, let the channel bandwidths are measured in Mbps*,* between each of these servers to *S* be denoted as $bw_{,i,}$ i= 0,1,2. Let these quantities be $bw_o = 3$, $bw_1 = 2$, and $bw_2 = 1$, respectively. Thus, with our definition, in this example, bw_o is the fastest channel; bw1 is the next fastest and so on. Hence, sending a unit load on bw_o takes less time to reach *S* than from others. We assume that when the server *S* receives the document from another server, it starts the playback simultaneously at the user terminal. After locating the respective servers having the requested multimedia document (in this case servers 0 to 2), server *S* adopts the following strategy. From each server a portion of the entire document is retrieved and the parts are collected by S in a particular order. Upon receiving a portion from *So,* the playback is started at the user terminal. Let the playback rate (expressed in the same units as the *bwi's),* denoted as *Rp* be 1.5 Mbps (MPEG I stream). Now, the retrieval strategy is such that before the playback of this portion comes to an end, the next portion of the requested multimedia document is collected from S_1. This process is repeated for all the servers participating in the retrieval process. This strategy has some inherent advantages. Firstly, it retrieves disjoint portions from different servers and thus, minimizes the retrieval time. Secondly, the strategy inherently takes care of continuity requirements, which are crucial when implementing such a strategy on network based environments.

Thus, the continuity in the presentation is one of the aspects that a MSR strategy guarantees in the retrieval process apart from access time minimization.

The access time is directly proportional to the size of the portion retrieved from server *So,* i.e., starting from the time at which the downloading starts to the time at which the playback starts. An immediate naive choice would be to make this size as small as possible to minimize the access time of the entire multimedia document. However, in that case we will later show that the *presentation continuity* cannot be guaranteed by choosing the first retrieved portion arbitrarily as small as we desire. Hence, using this strategy, the problem now is to decide on the optimal critical sizes of the portions of the multimedia document to be retrieved from each of the servers, satisfying the presentation continuity, using the bandwidth constraints, and the playback rate constraints to minimize the access time. In the above example, we see that the following size distribution m_o = 85.997 MB, m_i = 272.3235 MB, m2 = 665.679 MB satisfies the constraints, where m_i is the size of the data retrieved from server S_i, i = 0,1, 2, respectively. For this distribution, the access time (following the definition) is given by m_o/bw_o = 85.997 sees.

An elegant model of this retrieval strategy is by means *of directed flow graphs* (DFGs). Figure 4.4 shows the directed flow graph for this example. The communication nodes at the first level are assigned a weight equal to the total communication time of the portions of the multimedia document they are transferring to *S.* The dots on these communication nodes indicate that all these servers start their downloading simultaneously at time *t* units. Without loss of generality, we assume $t = 0$. If mo is the portion of the multimedia document communicated by S_0, then the total communication delay (which is the weight of the node 0) is given by cs_o/bw_Q. The second level nodes are referred to as *playback* nodes. The weight of the playback node 0 is proportional to the total time of playback of that portion of video, given by *mo/Rp,* where *Rp* is the rate of playback. The directed arrows depict the *causal precedence* relationships between the node events. Thus, at *S,* the display of the portion from S2 starts only after the display of the portion from S1 is completed and also the portion from S2 must be completely available.

4.2.2 Some Definitions

Throughout the chapter we will use the following definitions.

1. *Retrieval schedule distribution:* This is defined as an ordered tuple 'm' given by m = (mo , mi,...,m $_{N-1}$) where, mi is the portion of the multimedia document downloaded from server Si, i = 0,1,2,..., N - 1. Further, $\sum^{N-1} mk = L$

   and $0 < m_i < L$, i = 0,l,..N- 1

2. The *Access Time* or the *wait time* is defined as the time between the instant at which the servers start uploading their portions to the time at which the presentation starts. This is denoted as, AT(*m).* Typically, this is the time to access the first portion of the downloaded data, given by cso/*bwo,* where *bwo* is the bandwidth of the established communication path from *So* to *S.* Hereafter, we shall use the term "access time" throughout the chapter.

3. *Minimum access time* is defined as, AT_{min}. Thus, from the above set of definitions and the strategy illustrated in the above example, the objective is to minimize the access time by determining the optimal sizes of the portions of the video to be retrieved from different servers involved in the retrieval process.

4. The *critical size* , denoted as cs_i, i = 0, 1, 2, is the minimum size of movie that a client should retrieve before the playback of this portion could be started so as to avoid data starvation. This critical size depends on the available connection bandwidth of the channels and the playback rate of the movie at the client site [3].

4.3 Description of the Strategy

We now describe the details of the proposed strategy. Consider a scenario in which a requested movie is not available locally at the original server, denoted as, *S.* Without loss of generality, we assume that the requested movie is present at servers *So* , *S1* and *S2.* Let the total size of the requested movie be *L,* measured in *bits.* The connection bandwidths of channels from other servers (in this case So to S2) to the local server(in this case server *S)* are denoted as bw*i, i* = 0,1,2, measured in *bits per second.* Let the playback rate at the client site be *Rp,* measured in *bits per second.* Once locating the respective servers having the requested movie, server *S* starts retrieving the requested movie from these servers. Since the design of the Service Module supports multiple servers, the movie will be streamed from all available servers concurrently using separate connections.

This implies that the movie data will be partitioned into multiple portions and streamed from each of these servers. In this running example, a portion of the entire movie, denoted as mi, i = 0, 1, 2, is retrieved from each server and is collected by S in a particular order. Upon receiving the first portion of the movie from S_0, the playback may start at the user terminal, when retrievals from other servers are underway. As mentioned before, the presentation continuity is a major Quality of Service (QoS) requirement for a multimedia presentation.

Thus, in order to start the playback when retrievals from other servers are underway, the size of the portion retrieved must be such that there should not be any data starvation for playback. In other words, the size of the portion retrieved must guarantee the presentation continuity. Now, the retrieval strategy must be such that before the playback of this portion comes to an end, the next portion of the requested movie data should be made available from S_1. This retrieval process continues until all the movie is retrieved from the set of servers .Figure 4.2 shows the whole process of the above example

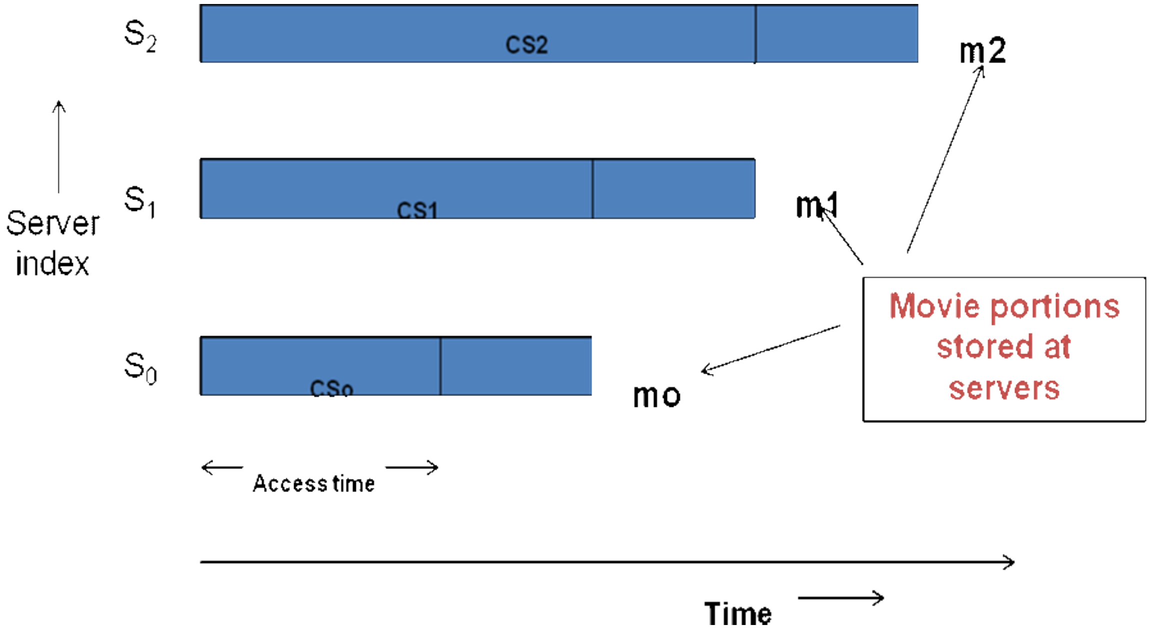

Figure 4.2 Schematic Diagram of the Retrieval Process

4.3.1 The Proposed Retrieval Strategy

Step: 1. Determination of critical size

Consider a scenario in which a portion of the movie of size '*m*' is to be retrieved from a server using a connection bandwidth 'bw' demanding a playback rate of *R*p at the client site. The client can safely start playing the portion after the critical size cs of this portion has been retrieved. In order to guarantee a continuous playback, the time to retrieve the remaining portion (m − cs) must be not greater than the entire playback duration of the portion m. In other words

$$\frac{m}{Rp} \geq \frac{m - cs}{bw} \qquad (1)$$

$$\Rightarrow cs \geq \frac{(Rp - bw) * m}{Rp} \qquad (2)$$

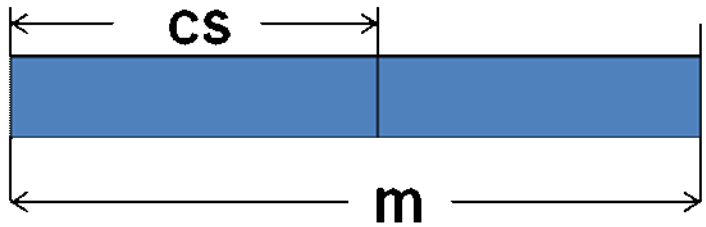

Figure 4.3 Critical size of a movie portion

Step : 2. Precedence relationship between retrieval and playback node

The retrieval process can be represented as a directed flow graph where the arrows capture the precedence relationships in the retrieval and playback portions. For example, portion i can be played after portion $(i - 1)$ and after receiving its critical size. From figure 4, we can derive a relationship between the retrieval of portion i and $(i + 1)$ and the playback time of the portion mi, with the use of the causal-precedence relation and continuity constraint as

$$\left(\frac{cs_i}{bw_i}\right) + \left(\frac{m_i}{Rp}\right) \geq \frac{cs_{i+1}}{bw_{i+1}} \qquad \ldots.(3)$$

By using the expression for csi from (2) in expression (3), we have

$$\frac{((Rp - bw_i) * m_i)}{Rp * bw_i} + \frac{m_i}{Rp} \geq \frac{(Rp - bw_{i+1}) m_{i+1}}{Rp * bw_{i+1}} \qquad \ldots.(4)$$

$$=> \qquad m_{i+1} \leq \frac{(Rp * bw_{i+1}) * m_i}{(Rp - bw_{i+1}) * bw_i} \qquad \ldots(5)$$

Let us denote $(Rp * bw_{i+1}) / ((Rp - bw_{i+1}) * bw_i) = \rho_i$. Rewriting (5), we have

$$m_{i+1} \leq m_i * \rho_i \qquad i = 0, 1 \ldots N - 2 \qquad \ldots(6)$$

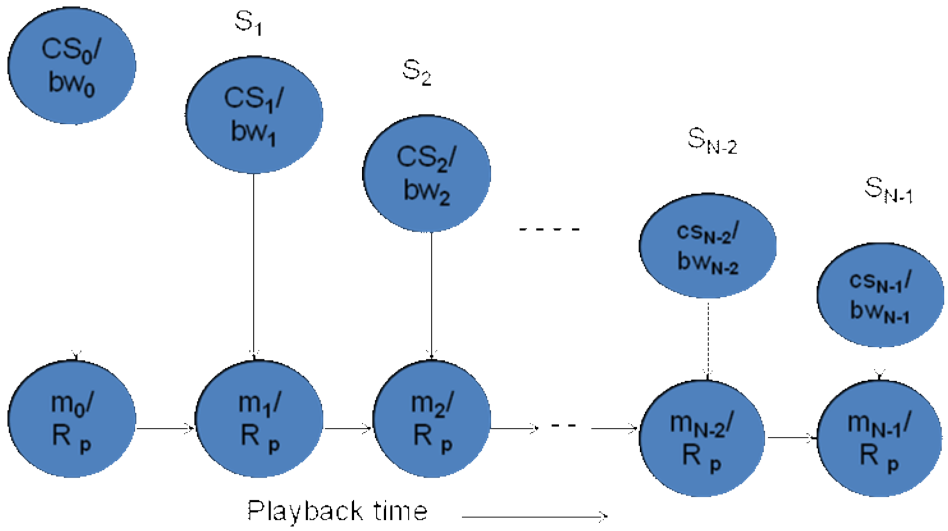

Figure 4.4 DFG for the retrieval Process

Step: 3. Determination of minimum access time

The use of equality relationships in (5) and (6) results in the maximum size of all the portions other than *m*0. But sum of portions stored at the servers is equal to the length of the movie. i.e.

$$\sum_{k=0}^{N-1} m_k = L \qquad \ldots.(7)$$

Hence, using (7), we obtain a minimum value for *m*0, equivalently the minimum cs0. In other words, we obtain a minimum access time. Each *mi* can be expressed in terms of *m*0 as

$$m_i = m_0 \prod_{k=0}^{i-1} \rho_k \quad , i = 1, 2 \ldots N-1. \qquad \ldots. (8)$$

Thus, the above set of (*N* − 1) equations given by (8), together with (7), are solved to obtain the individual disjoint portions of the requested movie. Substituting each *mi* from (8) into (7), we obtain

$$m_0 = \frac{L}{1+\sum_{p=1}^{N-1}\prod_{k=0}^{p-1}\rho_k} \quad \text{…. (9)}$$

Substituting (9) in (8), we obtain the individual sizes of the portions as

$$m_i = \frac{L\prod_{k=0}^{i-1}\rho_k}{1+\sum_{p=1}^{N-1}\prod_{k=0}^{p-1}\rho_k} \quad i = 1, 2 \ldots N-1 \quad \text{…. (10)}$$

Thus, the minimum access time is given by

$$AT_{min} = \frac{cs_0}{bw_0} = \frac{(Rp - bw_0) * m_0}{Rp * bw_0} = \frac{L\left(\frac{1}{bw_0} - \frac{1}{Rp}\right)}{\left(1+\sum_{p=1}^{N-1}\prod_{k=0}^{p-1}\rho_k\right)} \quad \text{…. (11)}$$

4.3.2 Performance Analysis

The performance of the retrieval strategy is evaluated by considering the case when the system handles one client request at a time and when the connection bandwidths are identical, i.e., bw_i = bw for all the channels. The movie size L is assumed to be 2 Gb, and the playback rate Rp is 1.5 Mb/s. Fig.5 shows an expected behavior of the access time with respect to the number of servers utilized, with a connection bandwidth of 1 Mb/s. From these plots, it is observed that the given strategy remarkably outperforms the PAR (Play After Retrieval) strategy [1] in minimizing the access time. Typically, with $N = 3$, the access time using PAR is 210.05s, whereas the access time using the given strategy is 52.51 s, yielding a significant gain of 75%.

No. of servers	Access time in Seconds (PAR strategy)	Access time in Seconds (Proposed strategy)
2	585.14	170.66
3	210.05	52.51
4	80.70	17.06
5	31.78	5.64
6	12.63	1.87
7	5.04	0.62
8	2.01	0.20

Table 4.1 access time Vs no. of servers

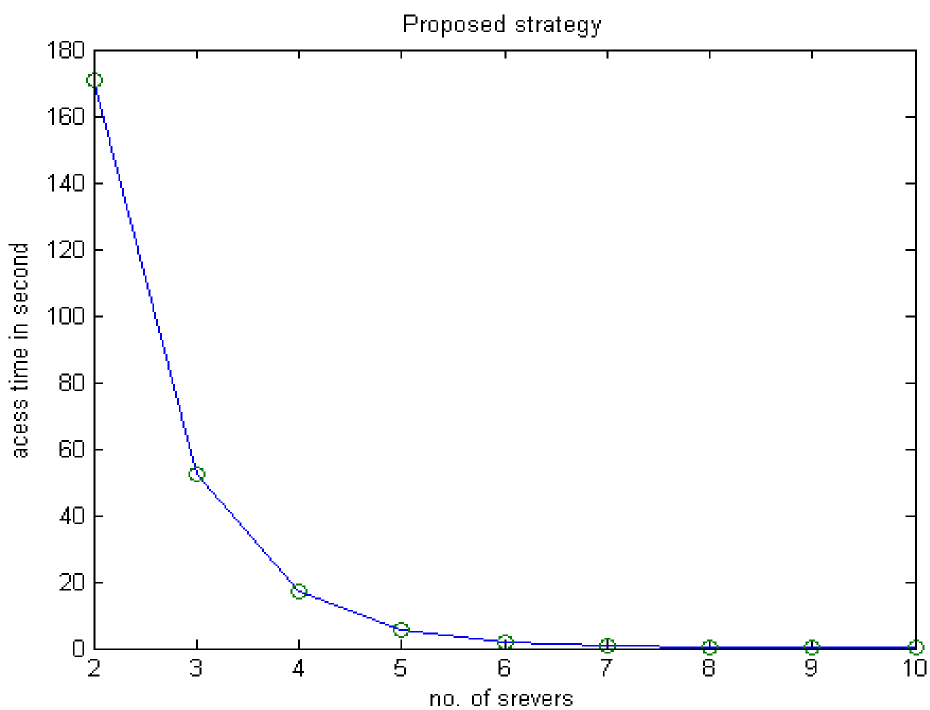

Figure 4. 5: Access time Vs no. of servers using proposed strategy

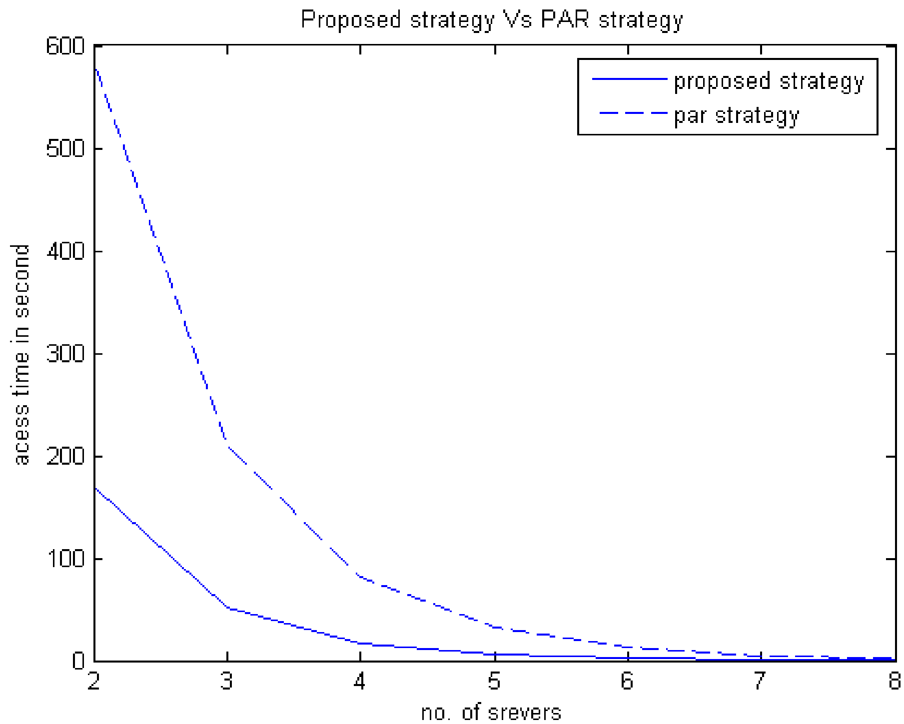

Figure 4.6: Performance comparison of proposed strategy and PAR strategy

4.3.3 Performance Analysis with server availability

The previous subsection discusses about the performance of the proposed retrieval strategy assuming that the servers are fully fault-free. This sub-section analyses the situation by considering the availability factor as a performance metric.

The availability *A,* the probability that a resource is in normal operation, depends on both reliability and maintainability of the resource. Thus, in general, the availability can be expressed as *Availability = uptime/{uptime+ downtime),* where *uptime* is the time duration in which the resource is in normal operation and *downtime* is the time duration in which the resource is not available for service, respectively. The availability for a constant failure rate and constant repair rate resource is known as steady state availability and is given as [29],

$$A_i = MTTF/(MTTF_i + MTTR) \qquad \ldots\ldots (1')$$

Where $MTTF_i$ is the mean time to failure of resource I and MTTR is the mean time to repair of resource i.

A resource can have an average availability ranging anywhere between 0 (low or no availability) to 1 (high availability). To ensure that the client receives the critical size cs_i of the media portion, say m_i, from S_i, the client starts its playback from time $cs_i/(bw_i*A_i)$ instead of cs_i/bw_i, thus allowing more downloading time, since $A_i < 1$, where A_i is the availability of resource pertaining to server *Si.* If this resource fails during the communication time period, the service provider will have more time to allocate any standby resources to recover the missing data or may wait for a failed resource to resume the communication. Thus, to guarantee a continuous playback under a realistic (failure prone) scenario, the time constraint relationship in the proposed strategy can be modified as,

$$\left(\frac{cs\ i}{bw_i * A_i}\right) + \left(\frac{m_i}{Rp}\right) \geq \frac{cs_{i}+1}{(bw_{i}+1) * A\ i+1} \quad \text{.... (2')}$$

$$m_i+1 \leq \left[\frac{(bw_{i+1})*(A_{i+1})*(Rp-bw_i)+bw_i*A_i}{(Rp-bw_{i+1})*bw_i*A_i}\right] * m_i \quad \text{... (3')}$$

The new ρi can be denoted as $\left[\frac{(bw_{i+1})*(A_{i+1})(Rp-bw_i)+bw_i*A_i}{(Rp-bw_{i+1})*bw_i*A_i}\right] = \rho_i$ ... (4')

Rewriting (5'), we have $m_i+1 \leq m_i * \rho_i$, i= 0, 1. . . $N-2$ ... (5')

So, the new data portions with server availability are given by:

$$m_0 = \frac{L}{1+\sum_{p=1}^{N-1}\prod_{k=0}^{p-1}\rho_k} \quad \text{..... (6')}$$

$$m_i = \frac{L\prod_{k=0}^{i-1}\rho_k}{1+\sum_{p=1}^{N-1}\prod_{k=0}^{p-1}\rho_k} \quad i = 1, 2...N-1 \quad \text{...... (7')}$$

From a client's perspective, knowing the failure and repair rates of a resource, the availability of that resource can be calculated using (1'). Based on the resource availability, the client starts the playback of each data portion from the time instant $cs_i/(bw_i*A_i)$ (assuming that the client makes a request at i = 0). Thus, by explicitly considering the availability of the server and its components, system *availability is* considerably improved.

Server index	Data size in Mb(PAR strategy)	Data size in Mb(Proposed strategy)
1	143. 9	3. 3
2	233. 9	8. 8
3	374. 2	23. 6
4	597.7	64. 1
5	956. 4	176. 0
6	1534. 3	487. 7
7	2467. 8	1362. 7
8	3979. 4	3832. 5
9	6431. 3	10840. 0

Table 4.2 Data size Vs Server index

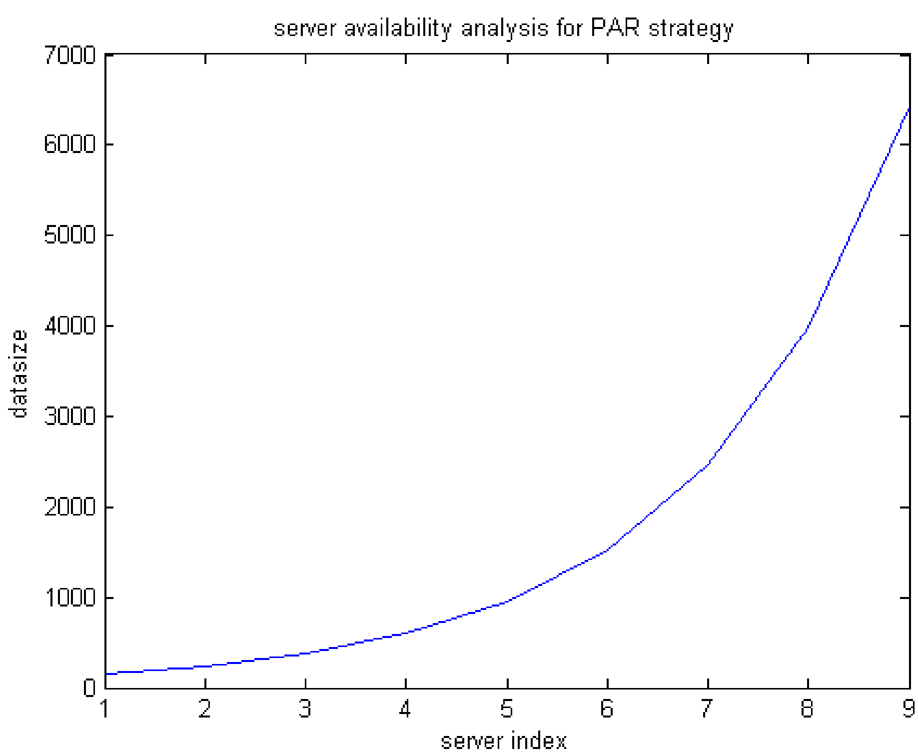

Figure 4.7 Data Size Vs Server index for PAR strategy

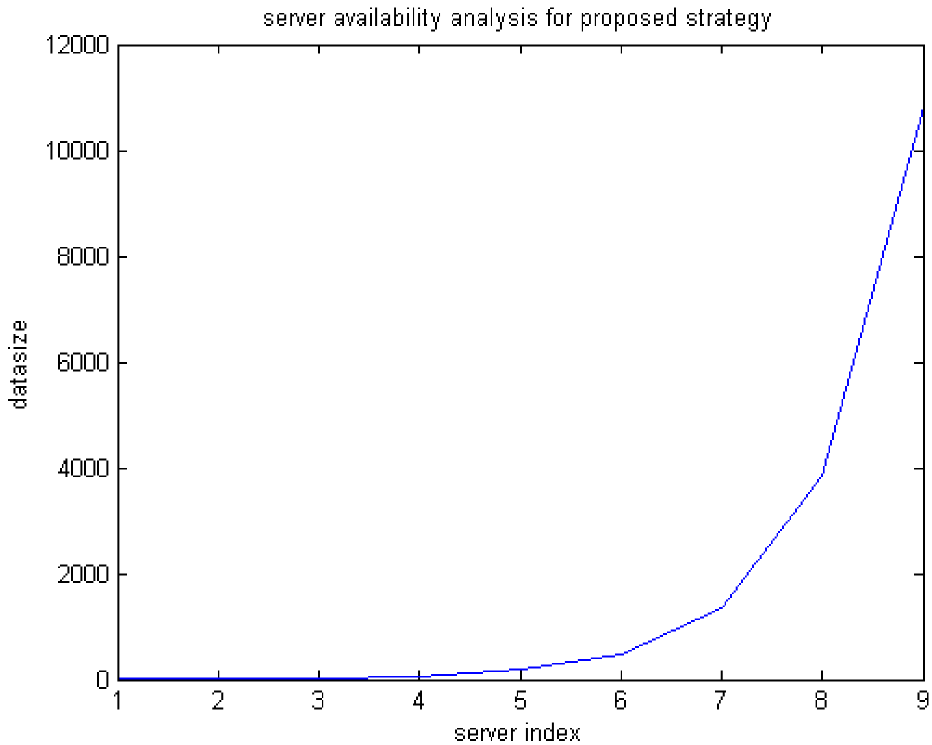

Figure 4.8 Data Size Vs Server index for proposed strategy

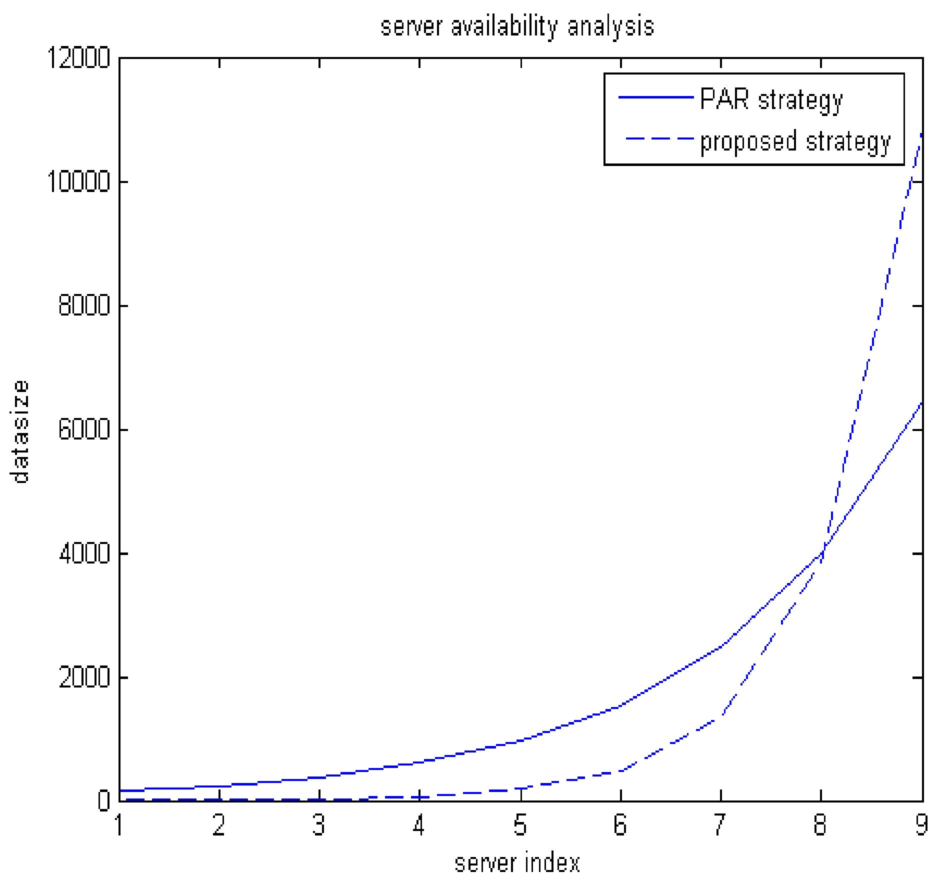

Figure 4.9 Comparison of Data Size Vs Server index

We compare the existing PAR strategy [1] with our proposed strategy by considering server availability. The plots are shown for a MPEG-I video streams with 1.5 Mbps playback rates, MTTF=1 hr for server 1,2hrs for server 2 and so on and MTTR=1hr for all servers.

From the simulation, it is observed that the servers with high availability will deliver more data (e.g. 10840.0 Mb for server 9) while servers with low availability will deliver less data (e.g. 3.3 Mb for server 1) to the client, a feature that is naturally expected in a frequently failure prone system [29]. Thus, the entire movie data are rescheduled according to availability of servers.

4.4Conclusion

In this chapter, we have presented a generalized approach to the theory of retrieving a long-duration movie requested by a client using a network based multimedia service infrastructure. For a network based environment, we have designed and analyzed an efficient strategy to minimize the access time of the movie. The performance analysis clearly highlights the advantages of the strategy when compared to the PAR approach which acts as the reference approach for our proposal. Also we have analyzed the strategies with the consideration of the availability factor. From the analysis, it is observed that the proposed strategy outperforms the referred strategy. There may be an increase in the access time when availability factor is introduced but it is more realistic because resources are not always available fully due to presence of some faults.

Chapter 5
Conclusion & Future Enhancements

Conclusion & Future Enhancements Chapter 5

There are some important issues that need careful consideration if a real-life system is to be developed on the basis of the proposed strategy. First, the transmission rates between the servers and the client should be known a-priori if the delivery schedule is to be calculated. This translates to the need to perform transmission tests in order to accurately estimate these bandwidths before any strategy is decided upon. Of course, this will consume additional time and generate more network traffic. An alternative approach might be to use the data of the requested movie in the tests, so as to compensate for such extra work. But this definitely complicates the retrieval strategy and requires some sort of schedule adaptation while the actual delivery is happening

The second concern is that the bandwidths must remain constant throughout the uploading procedure. Otherwise, the presentation continuity could be compromised. A third concern relates to the use of other techniques such as batching, patching, and caching, with the proposed strategy. Such a merge can lead to the conservation of the server and network resources, possibly allowing for higher system availability.

Further, taking into account the available bandwidth and buffer sizes at both the server and client sides, it would become more meaningful to combine the problem of admission control and retrieval strategy. This is a very critical problem when handling multiple client requests. In practice, most of the commercial players exercise software control for interactivity. While the interactions of Play/Stop/Pause/Resume can be easily incorporated in our scheme, other interactions, such as Fast Forward/ Rewind, Fast Search/Reverse Search and Slow motion, are major challenges to be considered.

Finally, we want to convey that the development in the thesis is genuinely supported detail literature survey and mathematics preliminaries leading to the proposed strategy. For the shake of continuity, each chapter has its relevant introduction ,theory and concluding remarks. The work is also supported by list of necessary references. Attempt is made to make the thesis self-content.

BIBLIOGRAPHY

[1] B. Veeravalli and G. D. Barlas, "Access time minimization for distributed multimedia applications," Multimedia Tools Appl., vol. 12, no. 2/3, pp. 235–256, Nov. 2000.

[2] D. Ligang, V. Bharadwaj, and C. C. Ko, "Efficient movie retrieval strategies for movie-on- demand multimedia services on distributed networks," Multimedia Tools Appl., vol. 20, no. 2, pp. 99–133, June 2003.

[3] Long Chen and Bharadwaj Veeravalli," Multiple-Server Movie-Retrieval Strategies for Distributed Multimedia Applications: A Play-While-Retrieve Approach", IEEE trans. on systems, man and cybernetics—part A: systems and humans, vol. 36, no. 4, pp. 786-803, July 2006

[4] Yiu-Wing Leung and Ricky Yuen-Tan Hou," Assignment of Movies to Heterogeneous Video Servers" IEEE trans .on systems, man, and cybernetics—part A : systems and humans,vol. 35,no. 5, pp. 665-681, Sept 2005

[5] D. N. Serpanos and A. Bouloutas, "Centralized versus distributed multimedia servers," IEEE Trans. Circuits Syst. Video Technol., vol. 10, no. 8, pp. 1438–1449, Dec. 2000.

[6] Chor Ping Low, Hongtao Yu, Jim Mee Ng, Qingping Lin, Yacine Atif," An Efficien Algorithm for the Video Server Selection Problem", Global

Telecommunications Conference(GLOBECOM '00) ,vol 3,pp.1329-1333, Nov.2000

[7] Hongtao Yu, Chor Ping Low, Yacine Atif," Design Issues on Video-on-Demand Resource Management",Proceedings of IEEE International Conference on Networks (ICON 2000), pp:199 – 203, 5-8 Sept. 2000

[8] Putra Sumari, Azman Samsudin, Hailiza KamaruIhaili," Data Storage and Retrieval for Video-on-demand Servers", Proceedings of the IEEE Fourth International Symposium on Multimedia Software Engineering (MSE'02), pp. 221-226,Dec. 2002

[9] Robert Flynn, William Tetzlaff," Disk Striping and Block Replication Algorithms for Video File Servers", International Conference on Multimedia Computing and Systems (ICMCS '96), IEEE Computer Society, Washington, DC, pp.105-109,1996,

[10] J. Aerts, J. Korst, and S. Egner, "Random Duplicate Storage Strategies for Load Balancing in Multimedia Servers," Information Processing Letters, vol. 76, nos. 1-2, pp. 51-59, 2000.

[11] Tat Seng Chua ,Cheng Hian Goh, Beng Chin Ooi , Kian-Lee Tan," A Replication Strategy for ReducingWait Time in Video-On-Demand Systems ", Multimedia Tools and Applications, 15, pp.39–58, 2001

[12] James Z. Wang, Ratan K. Guha," Data Allocation Algorithms for Distributed Video Servers", In Proceedings of the Eighth ACM international

Conference on Multimedia. MULTIMEDIA '00. ACM, New York pp:456-458, 2000

[13] Chor Ping Low," An efficient retrieval selection algorithm for video servers with random duplicated assignment storage technique", Information Processing Letters, vol.83, pp. 315-321, 2002

[14]A. Narasimhan, "Distributed multimedia applications-opportunities, issues, risk and challenges: a closer look", pp.455-460, 1997 IASTED International Conference on Intelligent Information Systems (IIS '97), 1997

[15] Blair G. Stefani, *Open Distributed Processing and Multimedia,* Addison-Wesley, 1997

[16] Hutchison D. Coulson G., Campbell A., Blair, Quality of Service Management in Distributed Systems *in Network and Distributed Systems Management,*Addison-Wesley, 1994

[17] Storey .N, Safety-Critical Computer Systems, Addison-Wesley, 1996

[18] Hutchison D., Mauthe A., Yeadon, "Quality of service architecture: Monitoring and Control of multimedia communications", *Electronics and Communications Engineering Journal* vol9, no3, p100-106, 1997

[19] Nahrstedt, K., Steinmetz, " Resource Management in Networked Multimedia Systems",*IEEE Computer* vol28 ,no5 ,May 1995

[20] Bochmann, G. V Hafid,"Some principles for quality of service management", *Distrib. Syst. Engng,* vol4, p16-27 ,IOP Publishing, 1997

[21] Koistinen, J., Seetharaman," A Worth-Based Multi-Category Quality-of-Service Negotiation in Distributed Object Infrastructures ",*HP Research Report HPL-98-51* (Hewlett Packard, 1998)

[22] Cristina Aurrecoechea, Andrew T Campbell, Linda Hauw,A survey of QoS architectures, ACM/Springer Verlag Multimedia Systems Journal, Special Issue on QoS Architecture, Vol. 6, No. 3, pp. 138-151, May 1998

[23] Xinping Guo, Colin Pattinson," Quality of Service Requirements for Multimedia Communications",Technical Report, Staffordshire University, 19th June 1997.

[24] Andreas Vogel, Brigitte Kerherve, Gregor von Bochmann and Jan Gecsei, "Distributed Multimedia and QOS: a Survey", IEEE Multimedia, Vol.2, No.2, pp.10-19 , 1995

[25]Infosys White paper, "Distributed video on demand-a grid based VoD solution", April 2006

[26]B.M.K Qazzaz,"admission control and media delivery subsystems for video on demand proxy server", Ph.D Thesis, April 2004

[27] Ernst Biersack, F. Thiesse and C. Bernhardt." Constant Data Length Retrieval for Video Servers with Variable Bit Rate Streams", IEEE *Proceedings of MULTIMEDIA '96,*pp-151-155,1996

[28] M. Ko and I. Koo," An Overview of Interactive Video On Demand System", Technical Report, The University of British Columbia, Dec 13, 1996

[29]B. Veeravalli,C.Chen and V.K. Prasanna,"Fault tolerant analysis for multiple servers movie retrieval strategy for distributed nultimedia applications", Journals of ,Multimedia tools and applications, Springerlink, Volume 32, No. 1, pp.1-27,January, 2007