Comunicazione Real-Time su Reti di Calcolatori: Introduzione ed Analisi dell'Approccio Tenet di Berkeley.

Pasquale Di Genova

9 December 1993

Contents

List of Figures

List of Tables

Premessa

Questo lavoro è stato eseguito all'Università della California a Berkeley (USA), presso l'International Computer Science Institute di Berkeley, ed all'Università di Bologna.

Lo studio è stato possibile grazie ad un progetto di scambio culturale fra l'ateneo di Bologna e le Università della California (di cui il laureando ha usufruito per l'anno accademico 1992-1993) e alla gentilezza del Prof. Domenico Ferrari (University of California at Berkeley).

L'era dell'informazione in cui viviamo oggi è caratterizzata da uno scambio di informazione facile e veloce attraverso i servizi di comunicazione diffusi. Nonostante ciò è pratica comune che a ciascun *tipo di media* vengano associati particolari tecnologie e servizi: il video arriva attraverso il servizio televisivo; l'audio ad alta fedeltà attraverso la radio; video e audio interattivi attraverso il telefono; i dati attraverso le reti di calcolatori. La tecnologia digitale può cambiare ciò attraverso l'unificazione dei modelli di servizio. Se tale unificazione avverrà e l'accesso al servizio sarà diffuso, allora l'efficacia e dunque anche l'importanza dei servizi di comunicazione diverrà piú evidente.

Molti ricercatori nel campo informatico e delle telecomunicazioni stanno trattando i problemi tecnici, risolti i quali, tale unificazione sarà possibile. I loro sforzi sono diretti a raggiungere conoscenze scientifiche e tecnologie ingegneristiche necessarie per costruire una rete capace di fornire sia vecchi che nuovi servizi (per esempio, video su richiesta e servizi di videoconferenza). Tale rete è chiamata Broadband Integrated Service Digital Network (BISDN), cioè rete a larga banda integrata nei servizi. Tale rete è stata anticipata da una rete digitale offerta dalle compagnie telefoniche e chiamata Integrated Service Digital Network (ISDN), cioè rete digitale integrata nei servizi. Servizi ISDN sono già disponibili ma la loro diffusione è lenta. ISDN non dovrebbe essere disponibile in tutte le case per molti anni ancora a causa degli alti costi di installazione e alla mancanza di servizi aggiuntivi che fornisce rispetto ai servizi *tradizionali*, Inoltre, è discutibile che BISDN possa diventare una realtà, almeno nella forma in cui essa è stata ideata oggi (cioè comunicazone a tutti, attraverso tutti i tipi di media). Infatti l'attuale tecnologia non è in grado di costruire una BISDN e, anche quando nel futuro la tecnologia ATM[1] sarà disponibile, bisogna ancora dimostrare che questa tecnologia sarà capace di fornire su una rete digitale una *qualità di servizio* pari o superiore a quella cui reti dedicate e sistemi di riproduzione, quali la televisione, la radio od la videoregistrazione, hanno abituato l'utente finale.

In contrasto, una rete diffusa e geograficamente estesa quale l'Internet, può essere impiegata per trasmettere video e audio, oltre al tradizionale traffico dati. Negli ultimi anni l'Internet mostra indici di crescita di tipo esponenziale[Pax93]. Tale crescita è imputabile ad un incremento del numero di host collegati alla rete, all'incremento del numero di applicazioni software che usano i servizi di rete, delle richieste di larga banda di nuove applicazioni che usano grafica, audio e video. Se gli attuali ritmi di crescita vengono anche solo mantenuti, presto le risorse della rete (banda di trasmissione, potenza di elaborazione nei nodi, potenza di commutazione) verranno saturate e dunque il servizio offerto diventerà sempre peggiore. Una possibile soluzione per garantire un livello di qualità rispondente alle richieste dell'utente, è la prenotazione di risorse. Questo approccio trova sempre piú consenso tra i ricercatori ed è l'approccio seguito dal gruppo Tenet, all'Università della California a Berkeley ed all'International Computer Science Institute di Berkeley.

L'oggetto della tesi di laurea è l'analisi di questo approccio.

[1]ATM è l'acronimo di Asynchronous Transfer Mode. Enti di standardizzazione hanno concordato che BISDN sarà implementato attraverso ATM.

General Introduction

This work was carried out at the University of California at Berkeley, at the International Computer Science Institute at Berkeley (USA), and at the University of Bologna, Italy.

The information era in which we live today is characterized by an easy and prompt exchange of information through widespread communication services. Nevertheless, it is still common that different media types be typically associated to different technologies and services: video arrives by the television service; hifi-audio by radio; interactive voice-audio by telephone; data by computer networks. Digital technology can change this by unifying service models. If such unification occurs and access to the service is wide-spread, then the effectiveness and thus also importance of communication services will become even more evident.

Many researchers in the computer and telecommunications field are addressing the technical problems to make such unification possible; their efforts are directed towards providing the scientific and engineering technologies necessary to support a network of both old and new services (for example video on demand and videoconferencing services). This network is called Broadband Integrated Service Digital Network (BISDN), and is preceded by a digital network supported by telephone companies called Integrated Service Digital Network (ISDN).

ISDN services are already available but their deployment is slow. Because of high installation costs and lack of additional services ISDN provides when compared to *traditional* communication services, ISDN is not expected to become available to all households for still many years. Furthermore, it is questionable that BISDN will ever become a reality, at least in the form it is depicted today (i.e., communication to all, through all media types). In fact, present technology cannot build a BISDN, and even when, in the future, ATM[2] networking technology will be available, this has to be proved capable of offering on a digital network the same or better *Quality of Service* that dedicated networks and end reproduction systems such as TV, HI-FI

[2]ATM is acronym of Asynchronous Transfer Mode. Standardization bodies have agreed that BISDN will be implemented through ATM.

radio, VCR, have already accustomed users to.

In contrast, a popular and geographically wide-spread computer network such as Internet[3] is being used to transmit video and audio, as well as traditional computer data. In recent years Internet shows exponential utilization growth rates[Pax93]. The growth is due to an increase in the number of host machines connected to the Internet, the number of software applications which use network services[4] and to the high bandwidth requirements of new applications that use graphics, audio and video[5]. However, if present growth rates are even only maintained, then soon the network resources will saturate and thus the service offered will degrade. Incrementing raw bandwidth is a temporary solution and contrary to what one might expect, if the introduction of high-speed links causes bandwidth mismatches in network nodes, then congestion can worsen (see section 1.5). A possible solution to service degradation and ensuring a Quality of Service is resource reservation. This approach is widening its consensus amongst network researchers and is the approach taken by the Tenet group at the University of California at Berkeley and the International Computer Science Institute in Berkeley.

The subject of the thesis is the analysis of this approach.

*

This thesis is divided in two parts.

The first part (chapters 1 to 3) is an introduction to the networking aspects of multimedia. We present the context of multimedia on high-speed networks, explain the new problems that arise and compare proposed solutions. The author feels that this background study is essential to both provide a framework into which fit new solutions in the rapidly changing area of computer networks, and to justify the need for new concepts and models. The background research also aims to familiarize the reader with today's technology so he can localize technological bottlenecks

[3]Internet can be considered as the union of all networks connected through the Internet Protocol. Today it counts 1,700,000 hosts in USA, Australia, Asia and Europe.

[4]Examples are distributed computer software such as Network File System (NFS), Remote Procedure Call (RPC), distributed databases which use network communication to facilitate resource sharing and reduce computing costs.

[5]for example desktop publishing, Visual Audio Tool (vat) and Network Video (nv).

and better understand the direction of future developments. In this sense we try to answer questions such as: is present technology sufficiently mature to support multimedia on current high-speed computer networks? Which high-speed networks are available in the world today? What does their architecture look like? What is the state of the art in protocols used in these networks? Which multimedia applications can be supported on today's packet-switched networks? The second part of the thesis (chapters 4-5) focuses on a possible solution to providing multimedia support on today's packet switched networks called the Tenet approach. This approach provides a more general realtime service, of which the multimedia service can be considered a subcase. In particular we describe a transport layer protocol called Continuous Media Transport Protocol, which is well suited for applications which generate traffic with periodic characteristics such as video and audio. At this level a simple algorithm has been introduced to optimize buffer usage. The algorithm has been implemented on a realtime network simulator developed by Edward Knightly, a member of the Tenet Group.

The thesis is distributed over six chapters.

Chapter one focuses on digital network technology and congestion. It familiarizes the reader to aspects of multimedia on computer networks, considers the evolution of the telecommunication network towards ATM, introduces high-speed networking, and finally explains the congestion control problem, motivating why this problem will become more important in high-speed networks. The conclusion of the chapter is the need for a realtime service model, a resource reservation model able to support a Quality of Service to multimedia applications.

Chapter two considers the state of the art in local area (LA) and wide area (WA) high-speed network architecture. Bearing in mind real-time applications, we examine LANs such as FDDI-II and DQDB. Then we consider a high-speed WAN called Xunet2 which spans the United States, from UCBerkeley, California to AT&T Bell Labs, New Jersey, other four high-speed networks also located in the United States, and finally a high-speed network oriented towards real applications called

Berkom, located in Germany, Europe.

Chapter three gives an overview of solutions to the realtime network problem. For example, computer network protocols such as ST-II, SRP and RSVP are examined and a new traffic control algorithm for packet-switched networks called Virtual Clock is explained. These solutions are compared to solutions adopted by the Tenet Group of University of California at Berkeley and International Computer Science Institute in their approach to realtime service in Wide Area Networks.

Chapter four focuses on the Tenet approach to real-time service support on high-speed packet switched networks. We state the importance of the Quality of Service issue and give a brief description of the tests performed at channel establishment time by the Tenet real-time protocol suite.

Chapter five considers one of the two transport protocols of the Tenet suite, Continuous Media Transport Protocol (CMTP), suitable for transmission of media streams such as video or audio. We implemented at this layer, on a real-time network simulator, called Galileo, an algorithm which improves buffer usage.

Finally, chapter six outlines future trends in computer architectures, in computer networks and in the services that they may provide.

Chapter 1

Overview of digital high-speed networks and their multimedia applications

1.1 Introduction

The full potential of multimedia applications can be exploited when using computer networking technology. In order to support such applications, new communication service models are needed. For example, a realtime communication service can support a superset of multimedia applications, because it provides a service with guaranteed performance[1]. Performance guarantees are the basis for a service that can ensure no packet loss and on time delivery of packets.
The first chapter provides the multimedia types that a computer network can transmit. In section 2, we describe multimedia types and their requirements, and give a picture of current multimedia technology. Section 3 explains the evolution of the telephone network towards BISDN, and introduces its underlying technology, ATM. Section 4 focuses on high-speed networks. We argue why more and more future networks will be high-speed, and suggest the importance of high-speed local area

[1]Performance indices are productivity, measured in throughput; responsiveness, measured in delay; and utilization.

networks to drive high-speed internetworking. Finally, section 5 defines the congestion control problem and examines some proposed solutions which however or do not solve congestion, or only predict network performances based upon the behaviour of other network users. The definition of realtime service, given at the end of the chapter, paves the path towards a two-party user-network model in which network performance can be guaranteed[2]. In this model a contract is stipulated between the network and the individual user as opposed to a model which considers the network on one side and the collection of all clients on the other, and where the behaviour of one client can degrade the service the network can provide to other clients.

1.2 Multimedia on Computer Networks

We describe the multimedia types, their storage and communication requirements, and examine current state-of-the-art technology.

1.2.1 Multimedia types and their requirements

Since humans naturally perceive information visually, auditory, and as text, to facilitate user-interface, applications may require the integration of the different media types. This requirement is called multimedia requirement of communication.

Although multimedia terms are becoming more and more common, no precise terminology in the area of multimedia has been provided uniquely within the standardization bodies[3].

A somewhat vague definition of multimedia is:

The term multimedia *is generally associated with information represented in a form receptive to humans, where the different kinds of information representation are referred to as* representation types *and may be character text, images, geometric graphics, audio and video.*[Egl90]

[2]The issue is developed in chapter 4.

[3]Some precise terminology is under development currently (i.e., ISO/IEC JTC1 SC2/WG8).

A brief taxonomy of representation types or media types is useful and follows:

1.Encoded text: characters represented by an agreed code (e.g. ASCII).

2.Still Images: can be vector graphics or bit maps.

2.1 vector graphics: a format in which images are stored as a list of instructions on how to redraw the image. For example, a line can be represented by a quintuple of the form

$< startx >< endx >< starty >< endy >< attributes >$

where attributes can be thickness, color, etc.

2.2 bit map images: a format in which an image is stored bit by bit. Since this format is a direct representation of the image itself, characters stored in bit map form are distinct from characters stored in an encoded text form.

3. Audio: can be digitized voice or digitized high-quality audio.

3.1 digitized voice: digital signals have a greater noise immunity than analog signals and a more strict bandwidth spectra. Thus if we compare the two at the same quality level, digitized voice storage is much more economic than analog audio storage.

3.2 digitized high-quality audio: digitized high quality audio provides quality comparable to that of consumer CD players.

4.Video: digitized video is obtained from an analog video signal source through the operations of sampling and analog-digital conversion.

Storage requirements

We now consider the storage requirements.

1 Encoded Text

Assume: two bytes per character, 140 characters per line, 66 lines per screen (these are typical values for a workstation), then

storage for one full page of encoded text requires:
$2 * 140 * 66$ bytes = 18480 bytes.

2.1 Vector Graphics

Assume: a *page* of vector graphics contains 700 lines [4], each page of 1024x1024 pixels, and 256 possible attributes (thus each line requires $4 * 10 + 8 = 48$ bits) then

storage for one page of vector graphics requires:
$700 * 48 = 33600$ bits.

2.2 Bit Map images

Assume a screen size of 1024x1024 pixels, 8 bits for each pixel color, then

storage for a bit mapped page = $1024 * 1024 * 8$ = 8Mbit=1MByte.

3.1 Digitized Voice

Assume an analog bandwidth of 4 kHz, 8 bit per sample, then applying Shannon's sampling theorem gives

storage for one second of voice requires:
$2 * 4000 * 8 = 64000$ bit/s.

[4] this would be a non trivial image.

3.2 Digitized High Quality Audio

Assume an analog bandwidth of 22kHz, 16 bit per sample, then

storage for one second of digitized high quality audio requires:

$2 * 22000 * 16 = 704000$ bits = 687.5kBytes[5].

4. Digitized Video

Assume PAL[6], encoded signal with a bandwidth of 5MHz, frame rate of 25 frames/sec, 24 bit per sample, then

storage for one digitized frame requires $\frac{10MHz}{24*25} = 1.2MByte$;

storage for one second of digitized video requires $25*1.2*10e6bytes = 30$ Mbytes.

Communication requirements

Interactive video and audio applications produce delay sensitive traffic. Bandwidth reservations that would guarantee an *uninterrupted* service for each media type are listed in table 1.1.

Table 1.1: Communication requirements of media types.

Media type	Guaranteed bandwidth required
Digitized Voice	64Kbps[1]
Digitized High Quality Audio	704Kbps
Low Quality Compressed Video	100Kbps
Medium Quality Compressed Video	1.5Mbps
Digitized Video	30Mbps[2]
High Definition TV	1Gbps or 140Mbps compressed

[5] 1kbyte = 1024 bytes.

[6] Phase Alternate Line is a color-encoding and decoding system for transmission of video signals, 625 lines per frame at 50 Hz. Other alternative coding systems are SECAM and NTSC.

[1] Telephone quality speech.

[2] Color full-screen full-motion video.

Table 1.2: Transfer rates and capacity of technologies.

Technology	Transfer Rate[3]	Capacity[4]	Adv/Disadv.
magnetic hard disk	$>$ 1MBps	100's of MB	Low bit cost
optical disk	$\cong$ 1MBps	1GB	Writeable only once
			high seek time
array of disks[5]	20MBps	7.5GB	Low cost
			very high transfer rate
non-volatile cache	15MBps	1MB	Good transfer rate
			limited capacity
flash memory	125kBps	128MB	Low transfer rate

Notice the demanding bandwidth requirement of digitized video. Because compression techniques can reduce required bandwidth, low quality compressed video can be provided at 100 Kbps; medium quality compressed video at 1.5 Mbps.

1.2.2 Multimedia technology: state of the art

We examine whether current technology can support the communication requirements of the media types described previously.

Current storage technologies

Continuous media types such as digital video and audio impose both a need for large volumes of storage capacity and sufficient data transfer bandwidth for retrieval of multimedia data.

Table 1.2 summarizes the transfer rate (read) and storage capacity of today's current technology.

How feasible is a simple video playback application (i.e., an application that displays stored video)?

[3] Read.

[4] B indicates Bytes, b indicates bits.

[5] Data from RAID-II project-1993 UC Berkeley, California. RAID is the acronym of Redundant Arrays of Inexpensive Disks. RAID-II presents a configuration of 24 disks of 320 MB each.

Comparing the transfer rates common storage technology offers such as hard disj or optical disk offers (a few MBytes) to the most demanding media type (i.e., digital video at 30MBps), we notice that the latter are in general one order of magnitude superior. New technologies like arrays of disks capable of transfering data at rates of 20Mbps, on one side and compression techniques capable of reducing the bandwidth requirement by a factor of 10, on the other, can bridge this gap. Thus in the near future, technology will be capable of generating data at speeds (20 Mbps) near to that requested by high bandwidth demanding video types (30 Mbps).

Current network communication technology

How feasible is a video display application through a network point-to-point connection? The quality of the service is subjective, but *slow* video at 128Kbps is possible if compression techniques are used. A tool that is capable of such video transmission is *Network Video* (nv), developed at Xerox PARC. Nv runs on a virtual network called MBONE built on the Internet.

It is necessary that network throughput be able to support the transmission rates specified in table 1.1 section 1.2.1. Table 1.3 summarizes the current network technology available, and its characteristics, whereas figure 1.1 summarizes bandwidth offered by networks and bandwidth required by media types.

1.3 Telecommunication Networks

1.3.1 Evolution of telephone network towards BISDN

Through the installation of digital technology, the telephone network is moving towards ISDN and trying to overcome its limit of being a dedicated network. Simulta-

[6]Workstations or phone-like equipment

[7]In the US.

[8]Primary rate service at T1 speed.

[9]Number of Pandora workstation boxes.

[10]CCITT G.703.

[11]Testbeds.

Table 1.3: Network technologies available

Network	Bandwidth(Mbps)	No.users[6]	geographical span(km)
POTS	0.0144	500M	telephone network
switched56	.056	-	telephone network
ISDN	0.144(2B+D)	250,000[7]	telephone network
ISDN	1.5[8]	-	telephone network
Ethernet	10	1024	LAN
Token Ring	16	250	LAN
Cambridge Fast Ring	50-75	20[9]	MAN
DQDB	45-150[10]	-	MAN
FDDI	100	500-1000	20
ATM	150-640	testbeds exist	LAN and WAN[11]

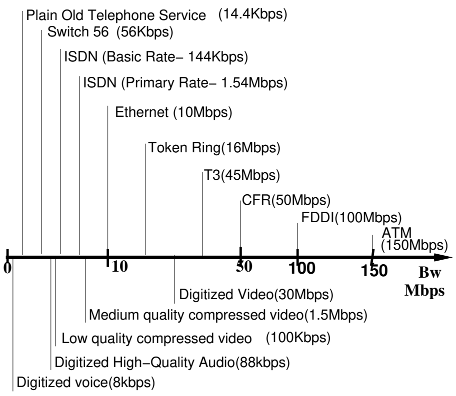

Figure 1.1: Summary Table: Above bandwidth offered by Network, below bandwidth required by media type.

Table 1.4: Evolution of communication networks.

Date	Technological Innovation	comm.bottleneck	Typical Bandwidth
1920's	copperwire link, human switch	link bandwidth	-
The introduction of microwave tx removes the link bottleneck and enables link communication costs to decrease more than the introduction of electronics can reduce cost of switches. The new bottleneck is switch bandwidth.			
mid 1960's	circuit switching	switch bandwidth	4kHz(analog)
Progress in microelectronics (the transistor and integrated circuits) improves switch technology and lowers their costs. The new bottleneck is link bandwidth.			
1970	packet switching	link bandwidth	64kbs(digital)[12]
Fiber optics, that provide a very reliable, high bandwidth medium are introduced in large scale. The new bottleneck is switch bandwidth.			
1980	fiber optics	switch bandwidth	100's of Mbps
New fast packet switching networks are being introduced (e.g. frame and cell relay).			

neously an increase in computer processing power, in link bandwidth, together with the development of new software tools are enabling multimedia applications to run on computer data networks. The development in electronic technology, in network protocols and in software applications is so rapid that it is difficult to understand what future networks will look like. However, its architecture will definitley depend on the costs of new technologies. Table 1.4 highlights this economical aspect, often not taken into due consideration. It shows the evolution of the telephone network through circuit switching techniques, to packet switching, to present day fast packet switching, evidencing the inventions that have enabled the transitions.

In the mid 60's **circuit switching** architectures were introduced. This architecture dedicates a physical line from the sender to the receiver for the entire call duration. This leads to an inefficient use of network resources due to burstiness of traffic. A traffic source is bursty when it is not continuously transmitting data, but instead there are time-intervals in which the data source is silent. To opti-

[12]Link bandwidth of ARPAnet, a data dedicated network born in 1970.

mize usage of the link bandwidth, the switch bottleneck is removed by introducing **packet switching** architectures (1970). In circuit switching architectures bandwidth is *statically* reserved in advance, whereas in packet switching the bandwidth is acquired when needed (i.e., transmission of a packet) and released after usage (i.e., after transmission). Because of this statistical multiplexing, network utilization increases. Packet switching also introduces routing that both facilitates reliability and reduces local congestion. Much of the effectiveness of packet switching networks is due to largeness of the networks in which this technique is used. In fact, studies show that efficiency of a network results from two, resource-sharing principles that apply when using packet switching networks:

1. law of large numbers principle.
2. economy of scale principle.

Because of point one, large numbers of bursty data sources collectively behave as one low-bursty traffic data source, thus improving resource utilization.
Because of point two, per unit cost of an item decreases for large production quantities. In packet switched networks we can allocate a 100 Mbps link and share it amongst 10 users, or allocate one 10 Mbps link for each of the 10 users. Because of the economy of scale principle, the first solution is cheaper.

Thus we can imply from the two principles that packet switching technology uses resources more efficiently because it enables many users to share many resources, rather than clustering groups of users to clusters of resources (point 1) other than being more cost efficient (point 2).

Optical fibers are the most recent technological innovation (1980's) and involve the communication media rather than the switching technology. Bandwidth costs have reduced dramatically; more and higher-speed bandwidth links are available.

How does new technology influence network design? Table 1.5 shows which functions are performed inside the network, for each network architecture[dP91]. It is evident that speed-up in the switching part of the network has pushed typical network functions (e.g., error control and error checking) to be performed at the network edge. Basically functions which can be guaranteed at the network boundary,

Table 1.5: Future network architectures will provide less complex services.

NET. ARCHIT/service	packet switching	frame relaying	cell switching
PACKET RTx	yes	no	no
FRAME DELIMITING	yes	yes	no
ERROR CHECKING	yes	yes	no

should be removed from inside the network, where the *same* function would be repeated *several* times (e.g. link error control is replaced with end to end error control).

Let's consider some examples:

-in a packet switched network, network layer and data link layer entities are in the network core;

-in a frame relay network[7], the network layer entities are moved to the network edge, whereas the data link layer entities remain in the network core;

-in an ATM network[8], some data link entities are moved to the network edge.

Because of this evolution towards BISDN, future networks will experience large time delays for correction or retransmission of data, which for example, in cell relay are performed on an end-to-end basis, rather than on a link-by-link basis.

1.3.2 Asynchronous Transfer Mode

Through use of video, audio and data types, BISDN aims to integrate all services on a single network: a digitized telephone network. Standardization bodies such as

[7]Frame relay is a data-link protocol. It defines how frames of data are assembled and rapidly routed through a packet switched network. Implementation of frame relay is in software.

[8]SMDS based upon ATM uses a cell relay architecture. Unlike frame relay in which variable length packets are transmitted, cell relay uses fixed size packets called cells.

CCITT have agreed to implement BISDN through ATM. In this section we give a brief introduction to ATM. Further material can be found in [dP91].

Definitions

The following definitions are useful. An authoritative definition of ATM, the acronym for *Asynchronous Transfer Mode*, is provided by CCITT (ITU) in Recommendation I.113:

> **ATM** *is a* Transfer Mode *in which information is organized in* cells. *It is* **asynchronous** *in the sense that the recurrence of cells is not necessarily periodic and that no rigid timing relationship is necessary between switches and hosts.*
>
> *A* **Transfer Mode** *is defined as aspects covering transmission, multiplexing, and switching in a telecommunications network. A* **cell** *is a* block *of fixed length and is identified by a label at the ATM layer. The ATM layer is the bottom layer of the BISDN protocol reference model. A* **Block** *is a unit of information consisting of a* Header *and an information field called payload.*
>
> *Finally, a* **Header** *is the set of bits within a cell allocated for functions required to transfer the cell payload within the network.*

ATM can be considered a packet switched connection-oriented protocol, in which packets are very small (53 bytes only) and are called cells. It can support both connection-oriented services (e.g., SMDS) and connectionless services (e.g., IP or Connectionless Broadband Data Service-CBDS[CCI80]). To adapt ATM layers to these services a convergence layer is needed. The intermediate service dependent layer, called AAL (ATM Adaptation Layer), together with the ATM layer can be

Table 1.6: ATM Adaptation Layers

Class A	Class B	Class C	Class D
required		not required	
constant bit rate	variable bit rate		
connection oriented			connectionless

considered equivalent to layer 2 of OSI reference model. ATM can support constant bit rate (CBR) and variable bit rate (VBR) traffic characteristics. According to the traffic characteristic and the connection mode of the upper layer (i.e., connection-oriented or connectionless), there are four different AAL layers[9]. These are shown in table 1.6.

Advantages and disadvantages of ATM networks

As we have seen, today's network bottleneck lies in the switching bandwidth; ATM networks provide high switching bandwidth (low switch latency) because of mainly three reasons:

1. it uses fixed size packets, cells, that simplifies hardware processing.
2. it is based upon a connection-oriented protocol [10].
3. it provides a connection oriented virtual circuit.

Some advantages of ATM are:

1. the network is simple.
2. the bandwidth allocation is flexible[11].
3. ATM can handle both continuous and bursty traffic.

Some disadvantages of ATM are [Ebr92, CS92]:

1. the network edges, i.e. the network-host interfaces, are complex.

---

[9] Recently a fifth has been introduced for computer data communication.

[10] Packet-switched *connection-oriented* service is the next best thing to circuit switching: routing is done only at connection-establishment time. In packet-switched *connectionless* service instead routing can be done on the fly, packet per packet.

[11] As opposed to STM, Synchronous Transfer Mode, where the bandwidth allocation is rigid.

2. end to end delays are not upper bounded.

3. there is no guarantee that cells will not be dropped.

4. ATM does not provide reliable delivery since it does not implement any retransmission or flow control mechanism.

5. the use of network resources is not optimal.

6. there may be bad network performance due to the small cell size[12].

7. ATM will be available only close to the end of the 90's.

8. The management of the ATM network still has to be defined.

Furthermore, due to the high variability of multimedia traffic sources, congestion can occur in ATM networks even if bandwidth is allocated[HS92]. It has not yet been decided how to guarantee a certain Quality of Service in an ATM network, although the ATM cell header includes a Grade of Service (GoS) field. A raw ATM connection can operate at a certain *Grade of Service* (GOS), but in reality the service provided is *best-effort* because cells with errors, or cells that encounter congestion are silently dropped. The retransmission of cells is the duty of the higher layer protocols at the network edge (for example TCP).

allocate resources. The problem is burstiness of traffic.

1.4 High-Speed Networks

Technological progresses in fiber optics communication media and the simplification of switches and the consequent increase of switching bandwidth enables networks to provide a higher transmission rate. Bandwidths of Mbps and above are available, on what can be called *high-speed networks*. No precise definition of high-speed network has been given. However, it is generally accepted that networks that can provide a bandwidth of 100 Mbps or superior be considered *high-speed*.

High-speed networks are being installed for three reasons:

1. To transport aggregate backbone traffic[13].

[12][KP92] shows, in the context of TCP connections, how performance can be improved when packets size is increased.

[13]A backbone network is a network that provides a communication link between two distant

2. Because new multimedia applications, such as graphics, audio and video require more bandwidth.

3. Because the bit transmission cost is much smaller[14].

1.4.1 Why high-speed networks are different

High-speed networks that span big geographical distances, to supply for example a backbone service, present a new characteristic: the network is a high capacity memory that stores the messages introduced onto the network but not yet received at the destination host. In fact, the quantity of information stored in high-speed WANs can in fact reach several megabytes. The bandwidth-latency product is a measure of the capacity of the network memory. For high-speed WANs, the network information content can reach several megabytes. Now we introduce some notation in order to estimate memory storage of different networks.

Let's consider a communication link of bandwidth capacity c and length l (see figure 1.2).

Let the bit source transmission rate be t_r.

For simplicity, let

$$c = t_r \tag{1.1}$$

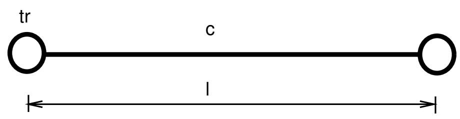

Figure 1.2: A communication link.

The memory capacity of the link is proportional to:

1. the number of bits introduced in the network per second, i.e. the source transmission rate t_r, which is equal to the bandwidth or the inverse of transmission delay.

---

networks, e.g., a FDDI ring that links two ethernets is a backbone network.

[14] Private 1.5 Mbps links, called T1 links, have exploded in the US for this reason.

Table 1.7: Networks as memories

Net	bandwidth	Span	Tx Delay[13]	Prop Delay	Memory
Ethernet	10Mbpss	2.8km	0.1 μ s	5 ms	50kbits
FDDI	100Mbps	100km[14]	0.01 μ s	1ms	100kbits
Xunet2	622Mbps	5000km	1.6 ps[15]	30ms	18.66Mbits

2. the time a bit takes to travel through the link, i.e., the propagation delay. Thus:

$$memory = TransmissionRate * PropagationDelay \tag{1.2}$$

Also, we remember that:

$$Delay = PropagationDelay + TransmissionDelay + QueuingDelay \tag{1.3}$$

where

• *propagation delay* is due to the finite speed of light (300,000 km/s in vacuum). Since the speed of light in an optical fiber is about 200,000 km/s, the propagation delay is:

$$PropagationDelay = \frac{1}{200,000\frac{km}{s}} = \frac{5\mu s}{km} = \frac{5ms}{1000km} \tag{1.4}$$

• *transmission delay* is due to packet length and for a one bit packet is given by:

$$TransmissionDelay = \frac{1}{linkbandwidth} \tag{1.5}$$

• *queuing delay* is the delay a packet experiences in a node while waiting to be served.

Table 1.7 gives, for different networks, the bandwidth, span, transmission delay, propagation delay and memory of a link of length equal to the span.

A consequence of the high memory storage is that end-to-end reactive flow control schemes are inadequate for high-speed networks. In fact, in case of congestion:

[13] Inverse of bandwidth.

[14] FDDI presents a dual concentric ring topology, thus the total fiber length is 200km

[15] 1ps = $10^{-9}s$

1. Too much data is sent before the receiver can detect congestion and inform the sender or upstream node to stop transmission. The amount of data lost is at least equal to the memory of the network link.

2. When the source is informed to stop transmission, congestion may have disappeared because it was temporary.

A possible solution to congestion is to have the *network nodes quickly* remove the congesting load. This approach, however, contrasts the trend of simplifying network core operations by moving functions such as network control functions to the network edges. An alternative solution is *network access control based on source rate control*: the user is permitted to transmit packets at a maximum peak rate. This approach considers the behavior of all senders, i.e. it is a holistic proactive approach, but it does not *prevent* congestion. Section 1.5 addresses the congestion control issue in more detail.

1.4.2 A brief classification of switches and links

In this section we define some recurring link speed terminology and give a classification of high-speed switches.

Link Hierarchy

Digital transmission of telephone quality voice occurs at a bit rate of 64Kbps. In fact, an analog voice signal presents a bandwidth of 4kHz. This can be converted into a digital signal without loss of information if we sample at 8kHz (Shannon's Theorem). If we consider 8bit per sample, we obtain a rate of 8*8,000=64Kbps.

We can bundle together 32 voice channels, and obtain a 2.048 Mbps link (CEPT-1) used in Europe, or only 24 voice channels and get a 1.536 Mbps link (T1 carrier).

Four T1 links give a T2 link at 6.176 Mbps; 28 T1 links give a T3 link at 43.232 Mbps. For higher capacity fiber optical links, a new family of transmission standards called Synchronous Optical Network (SONET) has been designed (see table 1.8).

Table 1.8: Transmission Rates

Industry Designation	Transmission Rate
DS0	64 Kbps
T1	1.5 Mbps
T3	45Mbps
Sonet OC-3	155Mbps
Sonet OC-12	622 Mbps
Sonet OC-48	2.4 Gbps

Switches

The goal of this section is to give an idea of the switch bandwidth and a brief classification of the switches currently available on the market [AD89]. A *switch* is a piece of a network that has the function of routing (see figure 1.3).

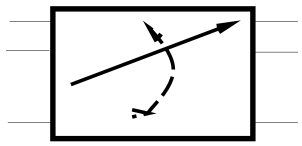

Figure 1.3: A switch's function is to route.

Packets arrive simultaneously at the switch input ports (on the left), are shuffled inside the switch and exit the output ports on the right. Packets designated for two different output ports may collide in an intermediate node inside the switch. This is called *internal blocking*. Instead, packets arriving on two different input ports, may want to be routed on the same output port. This situation in which two packets simultaneously require the same output port is called *contention*.

Based upon their internal structure, switches can be classified as:

1. Banyan and buffered banyan-based

2. Sort-banyan based fabrics
3. Fabrics with disjoint path topology
4. Crossbar-based topology
5. Time division fabrics
6. Fabrics with shared medium

1. In a banyan switch exactly one path exists between any input and any output (therefore the number of input ports equals the number of output ports); the switch is self-routing and can perform asynchronous, or synchronous operations if a buffer is provided (buffered based banyan). The switch may present internal blocking but is contention free. An example of a banyan switch is *ISPN* by Turner, that has a switching bandwidth performance of 1.5 Mbps.

2. In sort-banyan switches packets are presorted based upon their destination address, so there is no internal blocking. The first sort-banyan self-routing structured switch is the *Starlite* switch, that can reach a port speed of 150 Mbps.

3. In fabric switches with disjoint path topology, every input port has a direct path to every output port. This eliminates the internal blocking problem. Furthermore, the presence of an output buffer resolves the contention problem. An example is the *Knockout* switch, that provides a throughput of 32 Mbps per port and can handle variable-length or constant-length packets.

4. A crossbar switch presents a grid of connections between the input ports and the output ports. Only one packet may be present on an input bar at any given time instant, thus there is no internal blocking. The switch architecture is simple for small sized switches. Instead it is not suited to build economical large crossbar switches, since the number of connections grows quadratically with the number of input ports (equal to the number of output ports). An example of a crossbar switch is a *bus matrix* switch.

5. Time division fabrics can synchronize incoming packets in a common memory. Even though this switch works in packet mode, it operates very similarly to a conventional time division multiplexed (TDM) circuit switch. An example is the *Prelude* switch that can provide a bandwidth of 280 Mbps.

6. Fabrics with shared medium employ a bus or ring network architecture as switching medium. The bandwidth switching capacity is inferior to that provided by multipath switch architectures. An example of a switch built on fabric with shared medium is *Packetized Automated Routing Integrated System* (PARIS) that uses a high-speed shared medium bus to connect the input ports to the output ports. The presence of buffers and the usage of an arbitration protocol (which implements a fast token-ring algorithm) give the switch non-blocking propriety.

1.4.3 Are high-speed LANs the driving force for high-speed internetworking?

The importance of local high-speed networks for the deployment of gigabit broadband networks has been widely evidenced [Kle92, Fra91] and analyzed [LS92, FL91, Kun92] in literature.

An environment of high-speed local area ATM network islands connected by ATM WANs is technologically feasible [LS92]. The deployment of ATM technology can be accelerated if wide area ATM networks are preceded by local area ATM networks. This scenario is attractive because the telephone carriers, which are already investing in wide area ATM networks, will not have to pay the enormous installation costs for the last mile of connection (i.e., connection between the network and the end user). For a successful deployment of ATM technology, the cost of connecting a workstation to a local area ATM network need be carefully evaluated: a connection cost of $1000 seems to be an acceptable price to pay for a 20+MIPS (Millon of Instructions Per Second) $5000-$10,000 workstation. If ATM links become inexpensive, we will likely see equipment such as TV cameras and monitors provided with ATM plugs for direct network connection. With this possible development in mind, Xerox PARC has produced a complete design for an ATM LAN named BADLAN.

High-speed LANs are also being used to better understand high-speed WANs. High-speed local area (HSLA) traffic characteristics are analyzed in [FL91] and used to suggest congestion management policies for high-speed WANs. But what will the architecture of HSLA networks look like? Are there HSLA networks not ATM based? Switch based architectures seem to be a good choice [Kun92]. In fact, given today's computer architectures and their trends (see also chapter 6), a well-balanced computer architecture presents 1 MIPs of processing power for every Mbps of I/O data transfer; therefore, given today's availability of 100 MIPS workstations and 100 Mbps I/O devices[15] network point-to-point link bandwidth of 100 Mbps would well balance computer architecture design. A local ATM network can provide this[16]. ATM point-to-point networks also provide the following advantages:

1. good fault tolerance characteristics (rerouting is easy if a link fails).
2. greater aggregate bandwidth.
3. scalability (it is easy to add bandwidth capacity incrementally).
4. low latency since the architecture is switch-based rather than medium-shared based.

In summary, the deployment of BISDN driven by wide area telephone networks seems to be economically inconvenient for the service provider (the same telephone companies). Instead, HSLA networks capable of supporting multimedia applications can drive the need for internetworking. Furthermore HSLA networks can leverage from the popularity of local area networks which are already wide-spread, economical, technologically stable and capable of offering distributed applications.

Historically LANs have already been fundamental for the successful deployment of another wide area network, the Internet. This was born in 1969 to provide point-to-point *homogeneous* packet switching connectivity at 64 Kbps between campus and research center hosts throughout the US. After, Internet soon developed into a backbone network for connecting LANs on these sites[PT90]. A key of the success of the Internet is its capability of connecting *heterogeneous* subnets and a wide range of systems ranging from PCs to CRAYs. Like LANs drove the success of the Internet, high-speed LANs (HSLANs) may drive that of high-speed internetworking and in the future, LANs capable of supporting multimedia applications, that of a BISDN. If this will be the case, then one of the first services requested from a public packet switched network may be HSLAN interconnection and in the future, realtime service

[15] For ex. the DEC TURBOchannel.

[16] FDDI provides bandwidth of 100 Mbps but on shared media.

(see §1.6.4) on a packet-switched WAN.

1.5 Congestion Control

Any congestion control strategy must be fair, efficient and stable.

Fairness means that bandwidth is shared equally among users who require it. Efficiency means that nearly all network bandwidth is available to active users. Stability means that if the offered load increases, the carried load (throughput) does not decrease. For bursty traffic, stability can be guaranteed if the network can buffer all outstanding traffic (i.e., traffic non yet served). Finally, different traffic types require different quality of service guarantees. For example, voice and video require controlled delay and delay variance and packet loss.
Bursty traffic makes a network more congestion prone and avoidance of congestion more difficult. Because of trends in technology, traffic on communication networks will be highly bursty. Even continuous media traffic, that is traffic produced at regular time intervals, such as video, after applying compression techniques which reduces the bandwidth requirement by taking advantage of redundancies in consecutive frames, becomes bursty.

What does network congestion mean?

First some definitions related to congestion.

Definition 1: a situation of *congestion* arises when there is an *excessive* delay of packets or packet loss over a significant period of time[WC91].

What are the causes of congestion? R. Jain states that packet loss may be a symptom of congestion and not a cause in [Jai90]. He gives a definition of congestion related to resources usage:

Definition 2: a *resource*[17] is *congested* if in any time interval, the total sum of demands for that resource is greater than the available capacity.

[17] Network resources are link bandwidth, node processing power and node memory or buffer space.

Definition 3: *congestion collapse* is an *escalation* of a congestion, in which throughput is substantially reduced.

Definition 4: *congestion avoidance* is defined as the network's capability to signal to the hosts that congestion is occuring, and therefore that senders must decrease utilization of the network to avoid congestion [Jac88].

A definition of *congestion avoidance* can be based on the behaviour of network throughput [Jai90]. Here throughput, as a function of load, is modeled as:

$$throughput = \begin{cases} k * Load, & \text{if Load} \leq \text{L1} \\ k * Load1, & \text{if L1} \leq \text{Load} \leq \text{L2} \\ 0, & \text{if Load} \geq \text{L2} \end{cases}$$

and is shown in figure 1.4. The point Load=L1 is called the *knee* and the point Load=L2 is called the *cliff.*

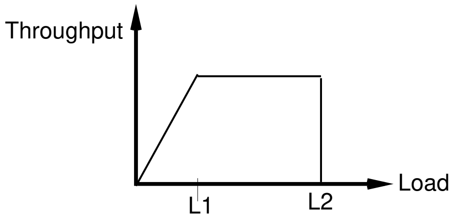

Figure 1.4: Throughput as a function of load.

Definition 5: a *congestion avoidance scheme* allows the network to operate at the knee.

Definition 6: a *congestion control scheme* allows the network to operate left of the cliff.

From this definition it is clear that congestion avoidance scheme is a particular congestion control scheme. Many solutions have been proposed to control congestion in a packet switched network. Before analyzing these, some myths need be clarified.

1.5.1 Myths on congestion control

Possible solutions to congestion can instead worsen the problem. If we simply consider the definition of resource congestion given in the previous section (definition 2):

$$ResourceCongested \longleftrightarrow \Sigma Demands > AvailableResources(inanytimeperiod). \tag{1.6}$$

then congestion could *always* be avoided by increasing the available resources. Installing bigger memories, higher speed links or faster processors seems to be sufficient. These solutions may increase congestion because:

1. Bigger buffers in nodes reduce packet drop but increase packet delay.
2. Higher speed links increase link-speed mismatch[18] and thus congestion at the lower speed link.
3. Like in 2, congestion can occur for high-speed processor mismatch.

In general, local congestion can be eliminated if we increase the available resources, but may congestion the network elsewhere.

Congestion control is a difficult problem because solutions are network architecture dependent. For example, different approaches need be taken if the network offers a connection-oriented service as opposed to a connectionless service. The policies that the solutions implement, i.e., the packet queuing and service policies, together with the packet dropping policy, also affect the congestion control scheme.

1.6 Proposed Solutions to Congestion

We will examine a congestion avoidance scheme used in the Internet, two predictive service approaches: one developed at Stanford University, California, one at MIT, Massachusetts; finally, we consider a guaranteed service approach developed at UC Berkeley, California.

[18] This will become more frequent in future when the gap between newly installed fast links and old slow links will increase.

1.6.1 Internet's solution: congestion avoidance

Van Jacobson in [Jac88] explains some new algorithms implemented in Berkeley's 4BSD TCP[19] which made congestion less frequent on the Internet, but does not completely eliminate congestion[20]. Basically, the approach is to avoid congestion by sensing on the network if there is a condition of congestion. If a condition of congestion is detected, the algorithm reduces the window size and thus the number of unacknowledged packets; if not, the algorithm increments the window size. Thus congestion control is flow control based and reactive. Congestion is not completely avoided and packets can still be dropped. The retransmission of dropped packets is done according to an exponential distribution).

1.6.2 Stanford University's solution: predicted service

Another solution is Stanford University's predictable level-of-service congestion control scheme: a rate based congestion-control algorithm which allows the network to provide a *predictable* level of service without requiring a priori resource reservation [WC91]. If the network load is known, and if the sender specifies its own transmission load, then through the use of network load-packet loss curves, a sender's packet loss probability can be predicted.

The load-loss curves are of the form $p_i(r_i)$ where the load is that traversing a gateway and

p_i is the probability of packet loss for sender i and r_i its transmission rate.

<u>Observation</u>: $p_i(r_1) \geq p_i(r_2)$ if r1 $\geq$ r2, i.e., the p_i(r) are monotone non decreasing in the load or transmission rate r: the higher the transmission rate, the higher the probability of packet loss.

In this case, the congestion control is source transmission rate based, but the network does not suggest an optimal transmission rate on the load-loss curves. In case of congestion, *which* senders should reduce their sending rates? By how much

[19] TCP is the acronym for Transmission Control Protocol, and provides a connection-oriented, guaranteed delivery service. TCP and IP form the Internet protocols suite.

[20] The study was undertaken after a congestion collapse in October 1988 on the LBL-UCBerkeley Internet link.

should the sending rate be reduced. In general, in this scheme *all* senders are told to reduce sending rates, and packets are discarded according to the loss-load curves p_i(r) of each specific sender; more precisely, packets may be discarded based upon two policies:

1. the load that the source host offers to the network

2. the traffic type to which the packet belongs. For example, in a situation of congestion, the policy may decide to favour loss-sensitive traffic and to drop loss insensitive traffic such as video and audio.

We now consider the two policies.

1. Rejection policy based upon sender's load

Let's assume that a host H_j initiates a new connection, starts sending packets and that these cause congestion. The rejection policy based upon load, discards sender's i packets with probability p_i given by the load-loss curve:

$$p_i = p_i(C, \tau_{total}, r_i) \tag{1.7}$$

where C=capacity of network; τ_{total} is the total network load; and r_i is the sending rate of host i.

How are the probabilities p_i defined? Two examples follow.

Example 1. No discrimination amongst senders.

$$p_i = constant, for - all - senders \tag{1.8}$$

Example 2. Punish only the sender with the highest sending rate.

$$p_i = \begin{cases} 1, & \text{for k so that } r_k = max\{r_i, i\epsilon\text{Set-of-all-senders}\} \\ 0, & \text{otherwise} \end{cases}$$

2. Rejection policy based upon traffic type

A second rejection policy is based upon the traffic type. In case of congestion, then loss-insensitive packets, such as packets transmitting video, are dropped rather than loss-sensitive packets (for example, packets of a file transfer). Thus the rejection policy is based upon the traffic type of the packets. This policy is not suitable when most of the packets involved in the congestion node are of the same type. In fact, in the extreme case in which all traffic were loss sensitive (for e.g., ftp traffic) then necessarily some of those packets would be dropped and thus the offered service is affected. In both policies the *punishment* is arbitrary, since there is not a clear definition of offending sender.
A second criticism of this approach is the amount of computation needed to draw the load-loss curves at each gateway of the network. The following simple example explains this.

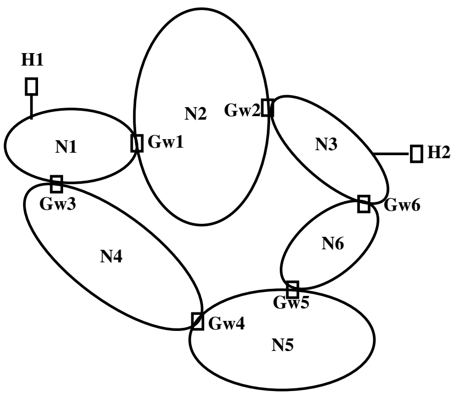

Figure 1.5: Each gateway GWi broadcasts its load-loss curves to all hosts Hj.

Consider the internetwork in figure 1.5 where hosts are indicated with H and gateways with GW.

The load curves are calculated by the gateways based upon the traffic that flows through them; these curves are then broadcast to all hosts[21].

Thus, in our example the gateways Gw1, Gw2,..., Gw6 broadcast to all hosts,

[21]In reality two parameters are used to represent the entire curve.

H1 and H2 included. Let's now consider H1 transmitting a packet to H2. H1 does source routing and can choose between path H1-Gw1-Gw2-H2 and H1-Gw3-Gw4-Gw5-Gw6-H2. Whichever path is selected, H1's sending rate is calculated using the load-loss curve for the most congested gateway on that path. Thus the highest loss probability of all the load-loss curves of the gateways of the route is selected (the bottleneck). The bottleneck gateway can be determined during connection-establishment. Then, at regular time intervals, each gateway recomputes the traffic that flows through it and broadcasts the curves to hosts so that, if necessary, gateway bottleneck information can be updated. Accordingly the sender can recompute its transmission rate on a certain path. Since global information of the network congestion is available at each host, source routing can be used to select the route which minimizes packet-loss. However, congestion is not prevented and can be controlled only through a reduction of the sender's transmission rate. In this sense, future service performance (packet loss) is *predicted* but not guaranteed.

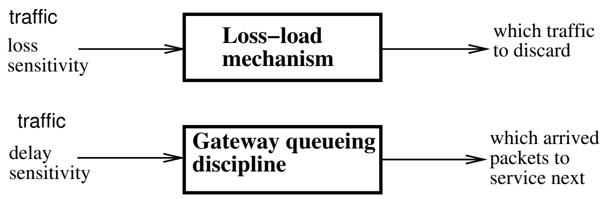

Figure 1.6: Scheme of loss-load mechanism and gateway queuing discipline.

In addition:

1. Each new customer degrades previous customers service performance.
2. The gateway can unilaterally deprive a sending host of its service.
3. There is no direct control of delay.

A refinement to the basic model explained above can be made to include delay control. In fact, as figure 1.6 shows, in general, the loss-load curve mechanism deals with the loss sensitivity aspects of traffic, and decides which traffic to discard (i.e. packet drop control); the gateway queuing service discipline, which implements an algorithm that allocates bandwidth, buffer space and promptness (delay), deals with the delay sensitivity aspects and decides which arrived packets to service next (i.e. packet delay control).

In conclusion, delay can be controlled through the service discipline adopted in the gateway.

1.6.3 Massachusetts Institute of Technology's solution: predicted service

New architectures and mechanisms are needed for real-time application support in an integrated service packet switched network [CSS92]. In addition, two new types of network services are needed: a predicted network service and a guaranteed network service.

MIT's solution to congestion in a packet-switched network is to provide a predicted service. In fact, for many real-time applications, a predicted service suffices [CSS92]. In a predicted service approach, the client gambles that future network performance is near to recent past network behaviour. The approach is based on the very strong assumption that the network conditions remain unchanged. Network traffic is measured and the measurements are used to decide whether to admit new traffic. The service is well suited for applications where a playback time can be associated to each packet and the display of the packet is independent from the real clock time. An example is video on demand service, through which a client can first request a movie stored in a remote movie server connected to the network and then play it back on its local host. In this sense, some flexible real-time applications can adapt to current network conditions and the traditional computer data transfer service. For example, the Internet, although does not support facilities for prescheduling resources nor facilities for denying service upon overload (and thus cannot guarantee quality of services to the network client) can still be used to transmit real-time applications that can move the playback point (so called adaptive applications). An example of an adaptive real-time application is VAT (Visual Audio Tool) used to transmit telephone-quality audio. But not all applications are adaptive. For example, interactive video and voice applications where the on time delivery of packets is fundamental.

If a network can ensure a predefined quality of service even in worst-case conditions, then it can support multimedia applications other than non-adaptive applications.

Three hard problems need be solved before a network can provide a guaranteed service:

1. to be able to specify and characterize the traffic to be served. Not only is traffic difficult to specify, but sometimes the specification is not reliable. In this case some kind of policing or traffic enforcement algorithm need be introduced.

2. to compute, a priori, the level of service that an application needs, and to map the characterization of this level of service to the service provider, that is, to the network.

3. to be able to upper bound delays and to know these bounds a priori.

If the network ensures a guaranteed service, and if the client conforms to its traffic characterization, then the service commitments will be met regardless of other network client behaviour.

1.6.4 UC Berkeley's solution: guaranteed level of service

From the proposed solutions to the network congestion-control problem, it is evident that ensuring a congestionless network service is not easy. A *realtime service* of a computer communication network is a service that can ensure, under certain conditions, that packets will not be dropped and that delays will be upper bounded. Therefore a real-time service is, by definition, a possible solution to the congestion control problem and is the basic underlying abstraction of ongoing research at the University of California at Berkeley and explained in detail in chapter four.

Here we give two definitions of realtime service selected from literature.

Definition 1: A *realtime service* can be defined with respect to ATM networks [dP91]:

A realtime service *is a service that requires that the bit stream arrives after a short delay at the receiving side of the communication link. ATM requires minimal functionality in the switching nodes, and therefore allows transport of packets at very high-speeds, thus* guaranteeing *very small delays at the receiver. [This implies] that due to the high-speed of ATM nodes in the network, it is possible to provide* realtime services *over an ATM network.*

This definition in somewhat misleading since it suggests that the high-speed characteristics of the network be sufficient to ensure a guaranteed service. In reality, a raw ATM network provides best-effort service.

Definition 2: A network independent definition of realtime service is [Fer90]:
A real-time computer communication service *is one whose clients are allowed to specify their performance requirements and to obtain guarantees about the fulfillment of those requirements.*

According to this definition, realtime communication can be considered a superset of network multimedia communication[DFZ92]. The definition summarizes the Tenet approach to realtime service in a packet-switched network; a unique approach in the sense that it guarantees performances as opposed to best-effort service offered by the current Internet and future ATM network, or a predicted network service suggested by Stanford University and MIT as their solution to congestion.

Chapter 2

Architectures in High-Speed Networks: the State of the Art

2.1 Introduction

In this chapter we analyze the state of the art in high-speed network architectures. We consider high-speed LANs, high-speed WANs, and the high-speed network experiences of the BERKOM project.

2.2 High-Speed Local Area Networks

The first generation of 802 class local area networks (Ethernet, Token Bus, and Token Ring) were designed primarily to provide asynchronous data service: synchronous and isochronous services have been an after thought, and in many cases are not realizable without significant modifications to the media access protocols[1]. In this section we consider two examples of LANs (FDDI-II and DQDB) which facilitate integration of synchronous and isochronous services and provide the means for transmission of media types such as video and audio. Synchronous traffic is traffic for which delays are upper bounded but random; isochronous traffic is traffic for

[1] There are plans for isochronous Ethernet services by adding a 6-Mbps dedicated virtual-channel to its 10 Mbps asynchronous service.

which delays are identical and not random.

2.2.1 Fiber Distributed Data Interface II

FDDI-II integrates high-speed packet and circuit switching services. It is the extention of FDDI, the last of the first generation of LANs. FDDI can transfer data at 100 Mbps, but as the power of workstations increases and thus also its I/O data rates, the FDDI network will be able to support fewer and fewer direct attachments. FDDI can extend up to 100 kms, but can handle only asynchronous and synchronous digital data traffic. Instead, FDDI-II is designed as a superset of FDDI, capable of delivering also isochronous services[Coo91][Ich91].

How is synchronous traffic served?

A sending station can transmit only when holding a token which rotates in the ring. The time a station must wait to obtain the token is upper bounded; the upper bound is called Time Token Rotation (TTR). If a new station wants to transmit, it must negotiate the request with the ring manager. If the ring manger accepts the new sender, then the TTR is increased. Isochronous traffic is served by a complicated mechanism based upon the subdivision of time in large intervals called super-frames. Each super-frame is partitioned in a time interval dedicated to the isochronous sources and a time interval dedicated to synchronous and asynchronous sources. Each sending station must be synchronized with the super frame, and a manager is responsible for assigning resources to the real-time traffic. The mechanism becomes more complicated when, for the transmission of a frame, more than one super frame is needed. Furthermore, a significant effort is required to interface FDDI-II to a public network. FDDI-II can serve both packet and circuit switching by a hybrid transmission mode. In addition, FDDI-II can provide the same cycle structure of the telephone network. To provide isochronous data services (such as video and audio) 125 micro second cycles are circulated on the ring.

A cycle consists of a SDU (service data unit) field containing 3120 data symbols and a PA (preamble) field filled with 5 symbols.

Because of the compatibility with the telephone network, FDDI-II can multiplex

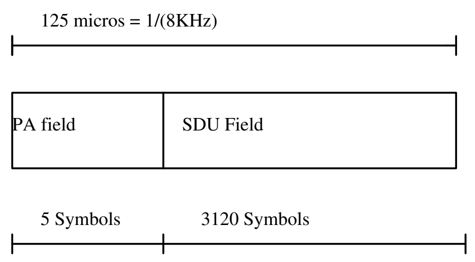

Figure 2.1: FDDI-II cycle structure

voice, video and standard data traffic. Data is originated from a directly connected host, or from a host connected through a intermediate LAN, and then transmitted over the public telephone network. For example, if a workstation were supplied with the equivalent of a T1 link (1.5 Mbps) over an FDDI-II link, voice could be transported over a 64 Kbps channel; compressed video over a 120 Kbps channel; the remaining 1 Mbps could be used for normal data transfer. A workstation to public telephone network through a FDDI-II multiplexer could look like figure 2.2.

2.2.2 Distributed Queue Dual Bus

The Distributed Queue Dual Bus (DQDB) is a computer communication network protocol based upon a slotted access mechanism. In a pure slotted network, slots are placed on the network by a frame generator, and captured by nodes that have data to transmit. The nearer the sending node is to the frame generator, the higher is the probability of finding a free slot. For this reason, pure slotted networks provide an unfair service. DQDB introduces a distributed virtual queue mechanism to reduces unfairness. The network presents a dual bus topology, where one bus is used for data transmission, the second for slot reservation. DQDB's reservation scheme works correctly (i.e., provides fair service) in the ideal case of 0 propagation delay. In the real case of non zero propagation delay, the slot reservation mechanism does not completely eliminate unfairness (i.e., bandwidth the network offers varies among nodes). The unfair bandwidth share depends on internodal distance. Let us consider the case of only two sending nodes. If there are many slots between

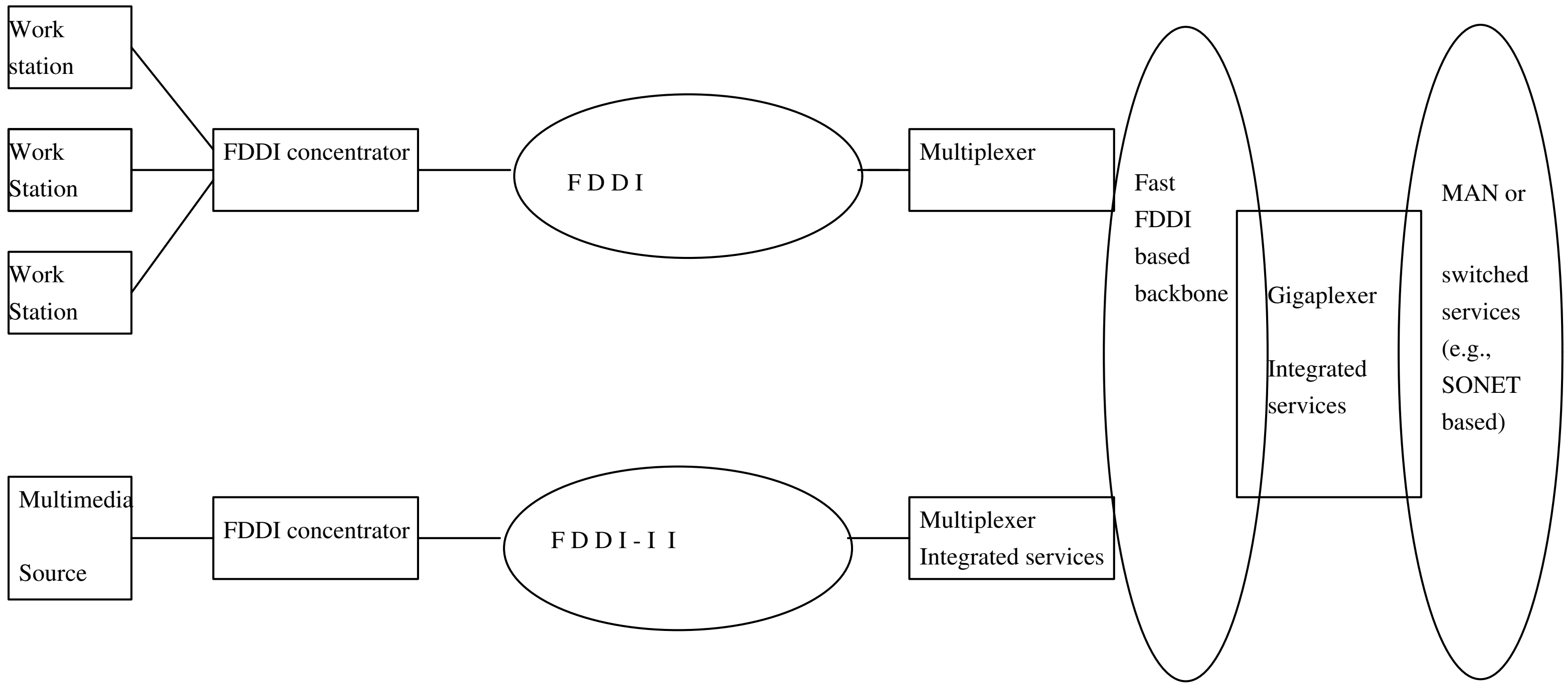

Figure 2.2: FDDI-II connection to a public telephone network

the nodes, the asymmetric bandwidth share can reach large proportions. Assume 2 sending nodes with data to transmit, distant 29 kms. This corresponds to 50 slots at a data rate of 100 Mbps. One sender may obtain 90This type of unfairness is called second grade unfairness. Because of unfairness, pure DQDB is not well suited to transport over long distances synchronous or real-time traffic such as video and audio. In fact, in first approximation, the protocol acts like a cyclic server with exhaustic discipline, i.e., when a node seizes the bus bandwidth, it retains the bandwidth until the node finishes to transmit all its data.

A consequence of bandwidth unfairness is message delay unfairness, that is the delay a message experiences in its delivery. High bandwidth throughput means low delays, which, because of the non uniform bandwidth share, corresponds to highly variable delays among senders.

This behaviour prevents a second node from transmitting data and obtaining packet delivery delay bounds.

The delay unfairness increases greatly for long networks, and thus because of the metropolitan extention of DQDB, is significant.

A more fair allocation of resources is possible when a rate control algorithm is implemented and network utilization is low. However, to provide synchronous services, and thus upper bounds on end to end delays, an algorithm for bus bandwidth reservation must be introduced.

Synchronous traffic accesses the medium via a Time Division Multiplexing (TDM) technique; asynchronous traffic through normal Distributed Queueing protocol (DQ). TDM presents the disadvantage that if no synchronous services are requested, resources are wasted. To serve real-time traffic, a modification of DQ, called general Distributed Queueing protocol (g-DQ), has been proposed [GP90]. The algorithm implemented in g-DQ deals with synchronous segments which access the medium via prearbitrated slots (PA) and a TDMA mechanism, whereas asynchronous segments continue to access the medium through the DQ protocol. However, even with g-DQ, further precautions must be taken into consideration to enable a correct service. For example, a bandwidth load balancing algorithm can be used to avoid that a single node hogs the network.

It is clear that providing a real-time service on DQDB is not easy.

2.3 High-Speed Wide Area Networks

To fully exploit the potentials of multimedia on computer networks, multimedia network support must be extended from a LAN to a WA environment. In this section, we consider five gigabit[2] testbeds, all located in the US, which have been built, or will be, to experiment on wide area networking. Sonet OC-12 speed represents the maximum bandwidth utilizable today (1993) for a single link. However, most of the current work is on networks with lower bandwidths (e.g., 155 Mbps which will be used for the telephone companies' BISDN service).

The networks connect three or four sites (industry research laboratories, universities and laboratories of the US federal government) through links of length from 30 to many hundreds of miles. Each testbed site is equipped with computers, switches and network equipment, which connect directly, or through LANs to the high-speed WAN.

The five testbeds are called: AURORA, BLANCA, CASA, NECTAR, and VISTAnet.
Below we give a brief description.

The AURORA testbed

The AURORA wide area network links, through three 622 Mbps SONET channels, four sites:
the University of Pennsylvania in Philadelphia; Bell Communications Research (Bellcore) in New Jersey; IBM's Watson's Research Center in Yorktown Heights, New York; and the Massachusetts Institute of Technology (MIT) in Cambridge, Massachusetts.

[2] 1 Gigabit=1e09 bits. Often very high-speed networks are referred to as gigabit networks even if the bandwidth offered is only of 622Mbps (Sonet OC-12). This bit rate is thus a "government" gigabit

AURORA is unique for two reasons: First, it will employ two different fast-switching technologies: an ATM switch and a switch based on Packet Transfer Mode (PTM), part of IBM's "plaNET" network architecture. This will provide a suitable environment to study issues that arise when underlying network technologies are different. Second, AURORA emphasizes experimentation with high-speed multimedia applications such as teleconferencing. The main research is on study of new paradigms for network resource control and allocation(MIT); study of distributed system paradigms such as distributed virtual memory (University of Pennsylvania); study of high-speed protocols, network architectures, and broadband network applications which all supply prototypes of ATM-based switching and transmission systems.

The BLANCA Testbed

BLANCA, a gigabit networking project, uses a ATM based network called Xunet2. Xunet2 links, through 45 Mbps links (T3 speed), five sites: AT&T Bell Laboratories in New Jersey, the University of Wisconsin at Madison, the University of Illinois at Urbana-Champaign, Lawrence Berkeley Laboratories (LBL), and University of California at Berkeley (UCB).

In future, sections of the network will be upgraded to 622 Mbps. Xunet network architecture is based upon key concepts adopted for BISDN, such as ATM and virtual circuit switching.

This architecture enables to support, at bit rates varying from 64Kbps to 45 Mbps, isochronous real-time voice and video, in addition to asynchronous data transmission. In isochronous mode, cells are transmitted at regular time intervals, and the network provides guaranteed bandwidth and controlled jitter. In case of congestion, isochronous traffic has a higher priority than other traffic types, whose packets are dropped.

Xunet2 is unique for two reasons. First, the sites are more distant than those in other networks (5000 Km separate New Jersey from Berkeley). Second, both traffic sources and service disciplines can be modified. In fact, the experimental ATM switches are designed so that researchers at the universities can implement their algorithms, change scheduling policies, and run experiments. The emphasis of experimentation is on distributed supercomputing, congestion control, the study of Internet protocols in an ATM network (U. of Wisconsin, U. of Illinois, and UCB), transport of mixed traffic (including video, audio, and data), and real-time communication (UCB).

Applications to be studied include remote visualization and control of simulations (U. of Wisconsin), medical imaging (LBL), and videoconferencing (AT&T, UCB).

The CASA Testbed

The CASA network connects, through a 800 Mbps network, four sites: the San Diego Supercomputing Center, Los Alamos National Laboratory, the Jet Propulsion Laboratory in Pasadena, and the California Institute of Technology (Caltech) also in Pasadena.

The main emphasis is on development of applications, which need high-speed networks to combine the processing power of multiple supercomputers (i.e., distributed supercomputing applications).

These applications are grouped in three categories:

1. computation intense modeling of chemical reactions

2. modeling of long-term global climate changes

3. interactive visualization of data describing the Earth's crust. This application requires access to remote databases, or requires more processing power than is available from a single supercomputer.

The NECTAR Testbed

The NECTAR testbed currently connects through a 100Mbps link, a LANs located at Carnegie Mellon University to a second at the Pittsburg Supercomputing Center. In the future, this connection will be upgraded to a 1Gbps, ATM-based link.

The main emphasis in NECTAR is on:

1. interconnection of high-speed local and wide area networks, focusing on design of a special interface circuit for HIPPI to ATM/SONET conversion.

2. study of improvements to design of computer architecture, which represents a bottleneck for data transfer to a high-speed network.

3. distributed supercomputing.

Some applications have been tested in local environments, and it is hoped that minimum changes are necessary to extend them to the high-speed WAN.

The VISTAnet Testbed

The VISTAnet testbed will link three sites: the North Carolina Supercomputing Center at Chapel Hill, the North Carolina State University's computer graphics research facilities, and the University's medical imaging research facilities. Although the initial testbed spans only a few tens of miles, researchers will develop specifications to cope with the many complex issues that arise because of network extention. The final network design will be based on proposed high-speed communications standards (SONET, ATM and HiPPI).The main emphasis is on distributed supercomputing, but VISTAnet differs from the other testbeds in its focus on a single application: a computation intense interactive display of 3-D images of radiation distribution. This application is used to help doctors plan cancer treatments based upon radiation therapy.

2.4 High-Speed Network Experiences of Berkom

What is BERKOM?

Berkom is a broadband communication project put into practice in the mid 80's in Berlin, Germany, with the support of the state of Berlin and Deutsche Bundespost Telekom (the Germany post and telephone company). To my knowledge it is the only broadband testbed in Europe. In this section, I will explain the scope of the project, the network architecture and the applications supported. In may be useful to bear in mind how the European communication philosophy and techniques differ from that of the United States.

European Network Technology	US Network Technology
connection-oriented on circuits	connectionless on shared-media
public switched	private networks
isochronous	asynchronous
national management	no central management

2.4.1 Scope of the project

Berkom is the only large project which promotes development of applications and end systems into a communication (telecommunication) context. It is unique in the sense that it provides a cooperation environment for research, industry, and users and supports a variety of broadband applications, reaching from the medical field, to distributed publishing, from information systems to multimedia.

2.4.2 Network architecture

The Berkom projects make use of two networks with optical fiber links:

1. Berkom Test Network (BTN): a 140 Mbps ring architecture network located in Berlin, and that uses two STM[3] and an ATM switch.

2. Vermittelndes Breitband Netz (VBN): a 140 Mbps fiber optical switched broadband network of the DBP [4] TELEKOM connecting 300 sites all over Germany. VBN is also known as *Videokonferenz Netz* because used to transmit analog video and audio. It is used for data transmission between computers that are equipped with a special network interface.

Table 2.1 summarizes the communication channels available in the two networks.

Berkom Test Network

Switches used.

[3]Circuit switched networks offer several channels of fixed bit rates in time division multiplexing technology and provide these data channels at different interfaces.

[4]DBP is the Germany post: Deutsche Bundespost

Table 2.1: Communication channels and the bandwidths available.

Network	Channel	Bit rate (Kbps)
Berkom STM	D,B1,B2	192
Berkom STM	H1	2048
Berkom STM	H4	139264
Berkom ATM	VC1	139264
VBN	Composite color video	analog
VBN	Audio (Stereo)	analog

1. STM switching system SEL-ALCATEL.

Based on circuit switching principle, the switch can support 64 Kbps ISDN channels, 2 Mbps and 140 Mbps channels. Can handle up to 32 access lines.

2. STM switching system Philips Kommunikations AG.

Based on circuit switching principle this switch or test network exchange can support bit rates of 64 Kbps, 2 Mbps and 140 Mbps. The number of subscriber lines is equal to 32.

Figure 2.3 shows the Berkom Test Network that includes the two STM switches.

3. ATM Siemens AG switch.

Can support 16 subscriber lines and a transport bit rate of 139.264 Mbps (per port). ATM presents the advantage that the end system can connect to the network through a single interface and that it can provide virtual channels at different bit rates.

Links used.

All links are in monomode optical fiber.

VBN

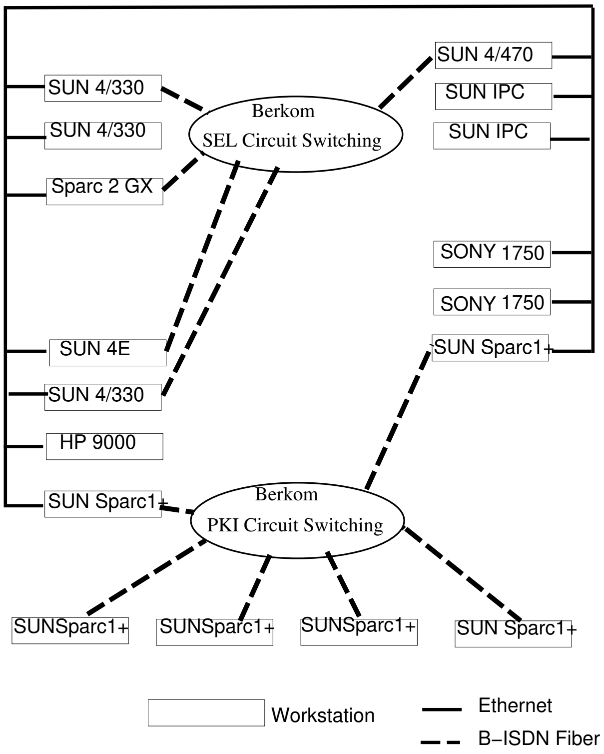

Figure 2.3: Topology of Berkom Test Network.

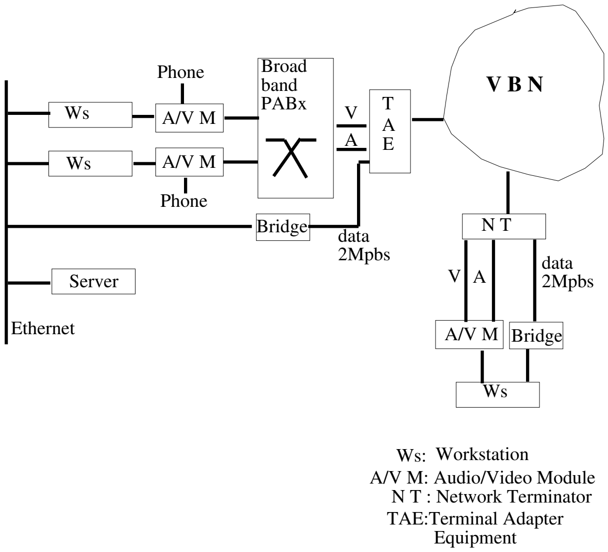

Figure 2.4: Audio, video and data connections to VBN.

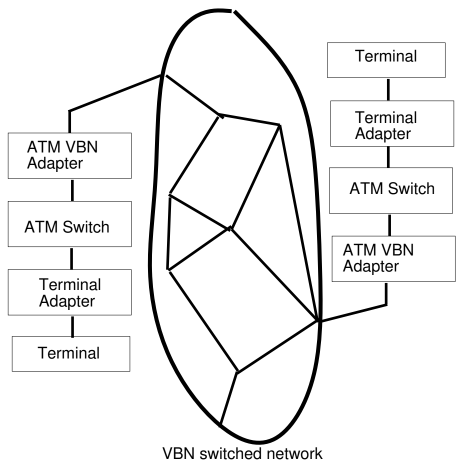

Figure 2.5: Interconnection through the VBN Network.

VBN offers transmission of video and audio according to tv quality standards, together with data channels at 64Kbps and 2Mbps. In many Berkcom projects the high data rate requirement of video is satisfied by 140 Mbps optical fiber links offered by VBN. Figure 2.4 shows how video, audio and data sources can be connected to VBN and figure 2.5 shows how two local ATM networks can be connected through VBN.

2.4.3 Protocols

The BERKOM Reference Model follows the ISO-OSI scheme and provides functions that can be classified as transport-oriented and application-oriented.
1. Transport-oriented functions: support reliable transfer of all data types between applications and end-systems connected via different communications networks.
2. Application-oriented functions: effect the control of the communication and of the presentation of information to be exchanged between application processes.

All applications projects, in the first instance, use TCP/IP protocols for data transmission, but future use of OSI protocols is considered.

To bridge the transition of TCP/IP to OSI, it is possible to temporary install both protocols stacks in the network nodes. The introduction of high-speed networks has moved the communications bottleneck from the network to the computer; more precisely, the bottleneck is the *protocol* processing[KP92] and its buffer data copying and management. Furthermore, bandwidth capacity is increasing more quickly than computer I/O transfer rates are. Thus the gap between network throughput and computer I/O throughput will widen and the bottleneck will become more evident. An example of an implemented transport protocol at Berkom is eXpress Transfer Protocol (XTP), a lightweight transport protocol; window management systems used are X11 and Motif.

2.4.4 Applications supported

Which applications have been experimented at Berkom?
Berkom has defined applications projects to offer services in *real* environments. These can be classified as:

1.Applications in Telemedicine.

2.Applications in Telepublishing Field.

3.Applications from the field of Broadband Information Systems and Multimedia (i.e. video).

A brief description of applications according to this classification follows.

Telemedicine

As can be deducted from the following project descriptions, the BERKOM experiences in Telemedicine focus on the service aspects of BISDN, and on the integration/connectivity of the end-systems to the network. To date (1992) the networks involved do not integrate all different media types. In fact a dedicated circuit switching network (VBN) provided by the German PTT, is used for video transmission; other European PTT carriers are providing dedicated communication links for videoconferencing applications. In this sense, because of the type of network used, problems of congestion have not been the key issue. An exception to this general rule is provided by the MEDAP project that focuses on medical imaging and where redundant servers have been introduced to avoid blocking and reduce transmission delay.

Project descriptions in Telemedicine

1. The *Medikom Project* supports medical conferences between two radiologist divisions that exchange X-ray images. To facilitate common diagnosis, digital data is transmitted together with audio and video, but two different networks are used to transport the distinct media types: X-ray images are transmitted over Berkom test network (BTN); video and audio over the switched broadband network, VBN.

2. The goal of the *Radkom Project* is to test picture archiving (X-rays) and communication between two clinics using BTN. Currently an ethernet substitutes BTN, and thus transmission time for one image is high (e.g., 20 seconds). Also, experience gained from this project shows that a traditional direct audio-video contact is helpful to guide the interpretation of the images.

3. The PAtient Data KOM goal is to provide computer supported intergration, communication and processing of patient data such as evidence reporting, image data and EGC curves. Communication is possible within the hospital and from the hospital to external partners through an ethernet and a DECnet, connected over TCP/IP.

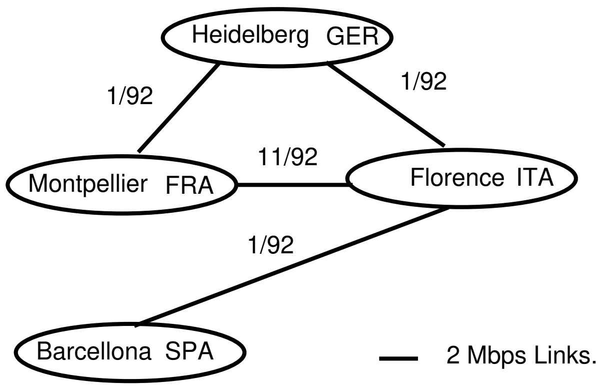

Figure 2.6: Video conference network used in RACE project TELEMED.

4. The *RACE project TELEMED* provides local and remote access of radio-

logical image databases, and transfer of patient data and images. Medical sites located in Heidelberg (Germany), Montpellier (France), Florence (Italy) and Barcellona (Spain) are connected by 2 Mbps reserved links provided by European PTT companies. The network topology is given in figure 2.6. A videoconferencing service is also available between the sites, that can achieve frame rates of 25 Hz and an image resolution of 128x128 pixels, 16 gray scales. The surplus bandwidth can be used for audio and data transmission.

Telepublishing

1. Telepublishing applications deal with design, editing and production of printing products. The goal of the *Joint Editing Project* is to provide broadband communication to facilitate retrieval of text, graphics, tables and photographs from remote sites and introduce them in a layout design offered by desktop publishing systems.

The configuration used is shown in figure 2.7. Communication is provided by BTN or LANs interconnected through TCP/IP. A consequence of this design decision is that for a participant to join the editing he need only establish a IP connection.

The project is based upon a client-server model and uses UNIX-OS[5] and X11 window system; the publishing environment is FrameMaker[6].

Application in Information Systems and Multimedia

1.The basic idea of *Broadband Information Systems* (BIS) is to support retrieval of multimedia information stored in a database (still pictures in high quality tv, audio information, text and graphics) and to make this information available in viewing points through use of standardized public and private networks.

A city information system based on this model is **Berlin-info**. This system has been in operational testing between 1987 and 1990 and is formed of six viewing terminals connected to a central database through one way 140 Mbps optical

[5]UNIX is trademark of UNIX Laboratories.

[6]FrameMaker is a trademark of Frame Technology Corp.

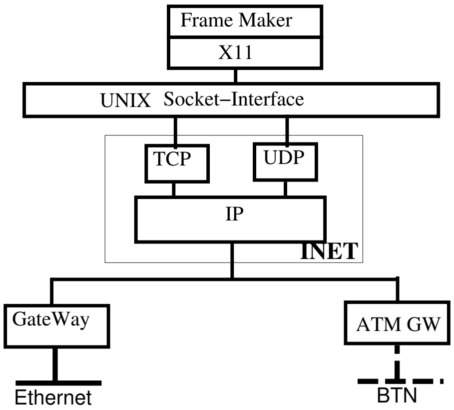

Figure 2.7: Configuration used in telepublishing application.

transmission lines using X.21 protocol.

2. An automobile service-oriented broadband information system enables remote training of technicians. VBN provides end users with 140 Mbps channels to support two-way videoconferencing service to the manufacturer's service sector.

3. Multimedia applications using client-server model. The approach taken in the *Office Broadband Communication project* (OBC) is to support retrieval of multimedia information from a server through a 1.4 Mbps link and provide it to a client[7]. The client can control the selection of the information through a low-speed control link to the server. Multimedia analog audio and video connections are provided by VBN network terminators and flow directly from the video disk players, on the server side, to a special video-mixing card in a workstation on the client's side. Digital data (text, graphics and audio) is instead stored on a WORM digital optical disk accessible through 2 Mbps channels.

[7]Client and server workstations are based on intel 386 microprocessor and use UNIX system V OS.

Chapter 3

Overview of Protocols and Solutions to the Realtime Network Problem

3.1 Introduction

In the networking community there is a general awareness that current Internet protocols are inadequate to support multimedia applications. The problem is burstiness of traffic and the challenge is to be able to deal with bursty traffic without blocking resources. Many research centers are developing protocols to support a new communication services in a packet switched network which ensures a *Quality of Service*. To provide such, new development is being made in the service model and network architecture (use of flow specifications, routing protocols, reservation protocols, admission control and packet scheduling). The interface the network offers to its clients (i.e., the transport layer or higher layers) is crucial for the effectiveness of the network. The DARPA Internet presents a datagram interface to its users as does SRP and RSVP, whereas new real-time models present a virtual circuit interface (e.g., ST-II and Tenet). We remember that datagram service is synonymous of connectionless service. Instead, virtual circuit service is synonymous of connection-oriented service and requires the establishment of a connection before data transmission. Hosts may choose to superimpose other higher level abstractions above the abstraction provided by the network. An example of Virtual Circuit on datagram is TCP on IP. An example of datagram on Virtual Circuit is IP on X.25. However, a protocol layer may provide a real-time performance guaranteed service only if all underlying layers provide themselves some guarantees [Cac89].

In this chapter we consider several solutions to the realtime network problem: ST-II, SRP, RSVP, Virtual Clock and the Tenet scheme.

3.2 ST-II: Experimental Internet Stream Protocol Version II

ST-II is a connection-oriented internetworking protocol for multiple-target communication that provides end-to-end service [LD93]. It incorporates the concept of stream that provides guaranteed or predictable communication characteristics required to support multimedia traffic over an internetwork.

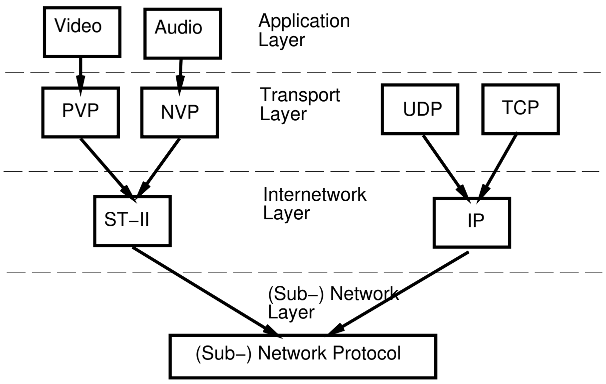

Figure 3.1: ST-II is an internetwork protocol suited for multimedia traffic.

3.2.1 Overview

ST-II[1] is the acronym for *experimental internet STream protocol, version II (the second)*. It is a protocol that provides end-to-end guaranteed and stream-oriented service abstraction upon a packet switched network such as, for example, the Internet. It can support both audio and video applications (see figure 3.1).

For example, Packet Video Protocol (PVP) and Network Voice Protocol (NVP), which run on end hosts. ST can run under TCP or UDP and coexist with IP of which it has a similar addressing scheme. ST entities need be installed in the nodes of the network, where they maintain state information for the streams that pass through it.

Stream state consists of forwarding and resource information. QoS requirements are expressed as a so-called flow specification (flow specs) or stream characteristics.

The main components of the QoS vector are:

- number of end points
- location of end points
- required bandwidth
- accumulated end-to-end mean delay
- accumulated end-to-end delay variance

which may be changed during the stream lifetime.

ST can support transmission 1 to m destinations, and recovery mechanisms for network failure. Compatibility with the internet protocols, in those subnets that do not have ST-II protocols installed is provided through IP encapsulation.

3.2.2 Communication model

ST-II is the core of the Heidelberg Transport System (HeiTS) for multimedia communication. The goal of HeiTS is to transport digital audio and video over a heterogeneous networks to meet their real-time requirements. HeiTS consists of three protocols (see fig.3.2): Heidelberg Transport Protocol (HeiTP), ST-II and Heidelberg Data Link protocol (HeiDL). Currently ST runs on AIX 3.2 OS on top of Token

[1] From here on, we will indicate ST-II with the abbreviated notation ST.

Ring LAN.

ST, more than a internetwork layer protocol, looks like a resource reservation scheme that assumes many mechanisms at its basis and uses other protocols to do typical networking functions: simple IP may be used to route packets on unicast connections; IP multicast may be used to provide packet delivery to more than one receiver, and TCP is used to ensure communication reliability.
However, ST-II does not determine how resource reservations are made, or how resources are scheduled according to these reservations.

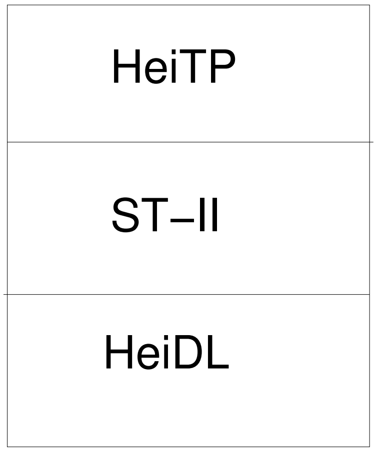

Figure 3.2: The Heidelberg Protocol Stack.

3.2.3 Resource reservation

Figure 3.3 summarizes the connection establishment phase: The client specifies its workload or flow specs, and the Quality of Service requirements.
The workload model is similar to the load model adopted in RSP where traffic is characterized by:

$$S = maxPacketSize \tag{3.1}$$

$$R = maxPacketRate \tag{3.2}$$

$$W = maximumworkahead \tag{3.3}$$

The last parameter allows for short-term violations of the rate R by allowing a burst of W packets i.e. a total of:

$$maxPackets = W + t * R \tag{3.4}$$

The quality of service requirements are function of data type and nature of application. They are:

$$QoS_{requirements} = \{D_{max}, \theta_{min}, reliability\} \tag{3.5}$$

where D_{max} is the maximum end-to-end delay, θ_{min} is the minimum throughput amongst all links of the multicast tree and reliability deals with error handling.

The ST-II resource reservation manager, HeiRAT, returns Quality of Service assurance on network performance.

End system resources are CPU time and memory whereas network resources are link bandwidth, switch processing power, and queueing buffer space[2].

Reservations may be made for an *optimistic* or *pessimistic* case.
In the pessimistic case (see figure 3.3 b), upper bounded peak values are specified and the QoS requirements are hard bounded. The performance obtained is called *guaranteed performance* service or worst case and it may lead to under usage of resources or over reservation.
In the optimistic case (see figure 3.3 c) average workload is specified and the QoS requirements are soft bounds.

The performance obtained is called best-effort service because no guarantees are made and because it may lead to a under usage of resources.

Bandwidth, CPU capacity and buffer space are *reserved* at channel set-up time but *allocated* only successively, upon demand, when a packet arrives in the node. Therefore, the availability of a resource is not guaranteed even after its reservation and thus a guaranteed service cannot be supported. To overcome this problem, ST proposed to adopt a resource probability function which returns the probability of successful allocation.

[2]These resources are associated to the network layer.

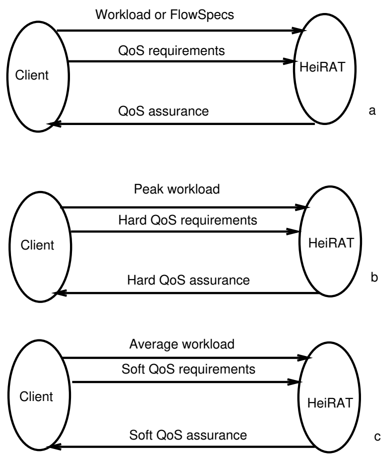

Figure 3.3: Connection establishment.

The resource probability returns the probability with which a reserved resource is available but it gives no information on future availability of resources since the resource probability at packet arrival need not be the same as resource probability at resource reservation time. Thus, there is no control that reserved resources will not be used by other streams, and thus not available for the new stream.
In addition, a stream's flow spec may be changed after establishment time: the changes may be made on the target set (i.e., the end nodes of the multicast tree) or on the network resources request. In the latter case, however, success of a stream change is high only when the new flow specs demand less resources.

3.2.4 The protocol

The ST-II service may be distinguished in a set-up phase and a data transmission phase.
In the set-up phase the route is selected and internetwork resources are reserved. The stream attributes (i.e., the multicast tree, allocated resources, state information) are updated. ST assumes that a route already exists and that the nodes that define it are returned from a function call. Hereby ST and routing are distinct and thus for this reason no routing algorithm is specified in the ST protocol. Static routing

seems to be a good solution given the resource reservation nature of the protocol. The routing function returns a set of next-hop nodes. A node consists of an ST agent and a resource manager. The latter determines whether resources are sufficient necessary to support the level of service requested in the Flow Specs. If so, resources are reserved.

A stream is set up after the participants of the communication have been specified (i.e. the data source, or origin, and the destination hosts or targets) and the appropriate transport layer protocol has been specified (i.e. PVP for transmission of video, NVP for transmission of audio).

Figure 3.4 shows the channel set-up interactions.

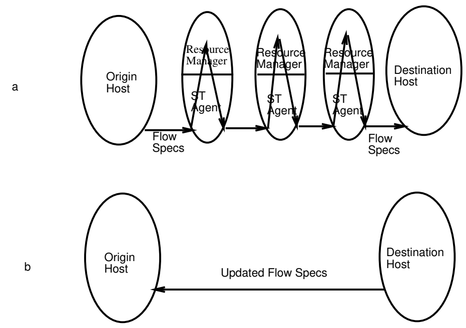

Figure 3.4: Channel set-up interactions: first round trip.

Figure 3.4 a shows how the origin host requests a connection reservation and specifys its Flow Specs.

In the reservation phase, ST agents exchange control packets of the *ST Control Message Protocol* (SCMP). (This protocol is also used to create, modify, and delete routing trees.) SCMP entities (present in the nodes of the multicast tree) negotiate the connection's QoS. As shown in figure 3.4, negotiation occurs between origin, targets, SCMP entities, and resource managers. The origin supplies *flow specifications* and a *requested QoS*. Each ST agent of the multicast tree receives these specifications as part of the channel establishment request, and forwards a resource reservation request to the local resource manager. The resource manager controls the usage of CPU time, buffer space and bandwidth on the outgoing link(s). ST-II does not determine how resource managers make reservations and how resources are scheduled according to these reservations, but it assumes these mechanisms at its basis.

Depending on the availability of resources, the flow specs may need be reduced as the reservation phase moves towards the destination set.

At the end of the forward pass, each destination host on the multicast tree, decides whether to accept or reject the final flow spec (Updated Flow Specs) which are thus retransmitted back up the origin (figure b).

In the second round trip of channel set-up, the origin considers the less stringent requirements amongst all the updated flow specs returned from destinations. It calculates the overall flow spec that all destinations can be provided with. In this second round trip reservations may be adjusted.

ST-II provides a *guaranteed* service in the sense that the network provides guaranteed service for the average data rate of each stream. Instead bursty traffic, that is not characterized in the flow specs, is serviced through reservation of additional network capacity shared among all streams.

Data transmission over ST-II is unreliable. It is implicitly assumed that packet loss is limited so as not to degrade significantly the quality of the received stream. This is a strong assumption.

Recently a network resource management called *Heidelberg Resource and Administration Technique* (HeiRAT) has been designed, which defines how resources are scheduled and reserved. A version on Token Ring LAN has been implemented [CVN87].

It provides media access through a rotating token and it upper bounds media access time by:

$$med.acc.time \leq N * T + \tau \tag{3.6}$$

where N is the number of stations on the ring, T is the time to transmit one packet, τ is the propagation delay. This mechanism is similar to the FDDI media access. HeiRAT also implements scheduling schemes such as Fixed-Priority (FP) and Earliest Deadline First (EDF). Finally, reliability is supported through modifications to Token Ring Protocol.

3.2.5 Critics to ST-II

The ST-II protocol leaves many points suspended and delegates many functions to other protocols. ST leaves open exactly how to reserve network resources and how to calculate QoS parameters. It assumes that some mechanisms are available for doing so, even if this is not trivial. For example, a straight forward way to guarantee on time data delivery is to set aside some bandwidth for each multimedia stream and to guarantee that this bandwidth is never used by other traffic. In this sense, bandwidth resource reservation is blocking. Even if bandwidth is not guaranteed, ST-II can still handle transmission of media types which can tolerate packet loss (e.g., video). However, ST-II does not guarantee that a given packet will not be lost, and thus is not suited, for example, to support a remote control service which requires delay bounds on packet delivery plus a reliable transmission.

Another limitation of ST is that the FlowSpecs must be the same for all receivers, i.e., a point to multipoint connection of *heterogeneous* receivers (i.e., receivers with different flow specs) cannot be supported.

3.3 SRP: A Resource Reservation Protocol

A second solution to the real-time network problem is Session Reservation Protocol (SRP). SRP is a distributed resource reservation protocol that allows performance guarantees to be made for communication based IP. It can be viewed as a network management protocol responsible for reserving network resources such as network bandwidth and CPU time.
No statistical performance bounds are offered to SRP's clients.

3.3.1 Overview

SRP schedules resources in hosts and gateways with the following properties:

1. SRP reserves CPU and network resources so that the protocol can guarantee communication performance.
2. The client must specify its workload which must be deterministic and time invariant.
3. SRP is transport protocol independent. For example, SRC may be used with TCP or new transport layer realtime protocols.
4. SRP is IP compatible.

3.3.2 Model and algorithm

The resource model assumes work is assigned to resources in discrete units called messages. Each message defines an arrival time (time to begin resource usage) and a completion time (time to end resource usage).

To describe message arrival, the following parameters are introduced:

S_{max} : maximum message size (bytes)
R_{max} : maximum message size (messages/sec)
B_{max} : maximum burst size (messages)

thus, the number of messages arriving at the resource in a time frame of width t is:

$$no.messages \leq B_{max} + t * R_{max} \tag{3.7}$$

where the parameter B_{max} takes into consideration short-term bursts of traffic.

Resource reservation

SRP is a resource reservation protocol where the client specifies its workload, the resource provides a bound on delay and then the client can decide if this delay is satisfactory for its needs.

RSP distinguishes two classes of sessions:

1. Guaranteed session
2. Best-effort session

In a guaranteed session, resources are reserved but cannot guarantee a QoS since resources can be overloaded. This may lead to message drop or excessive delay.

In a best-effort session no resource reservations are made.

3.3.3 The Session Reservation Protocol

SRP can be used on the Internet. It reserves network resources in the host and intermediate nodes.
The resource reservation protocol consists of five phases:

In an end-to-end session using SRP:

1. The sending and receiving clients set-up a transport level connection.
2. The sending client reserves local resources at its host.
3. The SRP module in the sending host examines the IP routing table to determine the next hop of the route and SRP reserves the appropriate network bandwidth resource and buffer space.
4. The SRP modules in each gateway on the packet's route reserves both CPU and network resources and corresponding buffer space.
5. The resource reservation phase reaches the receiving client, which compares the smallest possible maximum delay the network can offer to the sender's requirements; if the requirement is met, resources are reserved and eventual over-reserved resources are released.

After a session has been established, the sending client can begin to send data. IP datagrams arriving at the gateways are associated with the corresponding end-to-end sessions and the reserved resources are used.

3.3.4 Critics to SRP

1. An important claim of SRP is its compatibility with IP protocols. However, for the resource reservation model to work, the route must not change. Thus, the IP datagram service model has to be modified to provide static routing: IP, from being a connectionless service provider, becomes a connection-oriented service provider. Static routing can be achieved through a session call at the routing gateways.

This call specifies the next host to which all IP messages of that specific session will be sent.

2. A packet loss-free data service is not achieved, since there is no algorithm implemented to avoid overloading of resources.

3. SRP only establishes and deletes end-to-end sessions. No means to relax the requirements on such sessions are provided.

3.4 RSVP: A New Resource Reservation Protocol

A third solution is RSVP, the acronym for *new resource ReSerVation Protocol.*

RSVP is a connectionless internetworking resource reservation protocol. It can reserve resources on a unicast connection or on a multicast tree. Because a connectionless network service does not maintain memory, resource reservation is difficult. In fact, RSVP may have to reestablish resource reservations for each datagram which is routed on the fly.

A unique characteristic of RSVP is that the resource reservation is receiver initiated. This simplifies the reservation for heterogeneous receivers. If, in fact, a host is transmitting to multiple receivers, it must reserve resources on the corresponding multicast tree. If heterogeneous receivers are to be considered, the problem of how to distribute the reservation of the resources on the route in a sender initiated resource reservation is not trivial[Wid93].

3.4.1 Overview

Currently the protocol is at the design stage[LZ93]. Users specify their service requirements and the resource reservation protocol accordingly establishes and maintains resource reservation. To provide resource reservation on a connectionless datagram service, RSVP must gracefully deal with route changes. Resources are first reserved at channel set-up time, used during data transmission, and explicitly released when the channel is torn down. If the same where done in a connectionless service, for each datagram, the overhead would be high. Instead, in RSVP resources are automatically released if the reservation request is not refreshed. This approach is called soft-state approach.

3.4.2 Model and algorithms

RSVP provides a resource reservation scheme in a packet-switched network that for the *worst case*, is blocking. Reservations are made along routes to all receivers. The soft-state algorithm automatically releases unused network resources, facilitates adaptivity, and increases the robustness of RSVP. A disadvantage is the waste of resources which may be unused but not made available until the time-out interval expires. Because receives are responsible to refresh the reservation of resources, each sender maintains a sink tree. A sink tree is a tree that branches out from the receiver to all its senders. It is kept to reconstruct the reverse route of the data packets and deliver resource reservation messages to its nodes. Finally, because receivers are responsible for keeping the reservation active until data is received, the soft-state approach facilitates billing where the receiver would pay for data transmission.

3.4.3 Resource reservation

Resource reservation specifies what amount of resources is reserved for whom, where whom refers to *entities* that control the resources. The single resource reservation scheme does not define which *packets* can use the resources. Instead, a distinct function, called a *packet filter*, selects which packets can use the resources. Packet filtering enables to assign reserved resources to certain classes of packets, and thus provide different reservation styles.

RSVP can reserve resources for a point-multipoint connection. It can reserve resources even if not all paths from sender to receivers have same bandwidth capacity or desire the same QoS. In other words, RSVP can support heterogeneous client service requirements.

3.4.4 The protocol design

Let's consider the interactions between the application and RSVP at the sending node and the network (see figure 3.5).

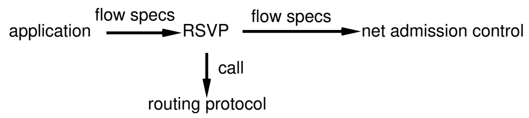

Figure 3.5: Interactions between application, RSVP and network.

Flow specs are specified by the receiver application and delivered to RSVP. This in turn forwards the flow specs to the network admission control blocks plus calls the routing function.

The flow specs have not yet been defined.
RSVP packets pass through the switches of the flows delivery path. (A path is defined by the sending node and the receiving node).

A switch may admit or reject the reservation request. If all switches of the route admit the flow, then the resource reservation is established. Thus, to maintain resource reservation , the receiver periodically sends refresh messages.

When a sender (or receiver) wishes to terminate the connection, the sender (receiver) sends out a path teardown message to release the reserved resources. Teardown messages are not retransmitted, because even if a teardown message is lost, the intermediate nodes will eventually time-out the corresponding state reservation.

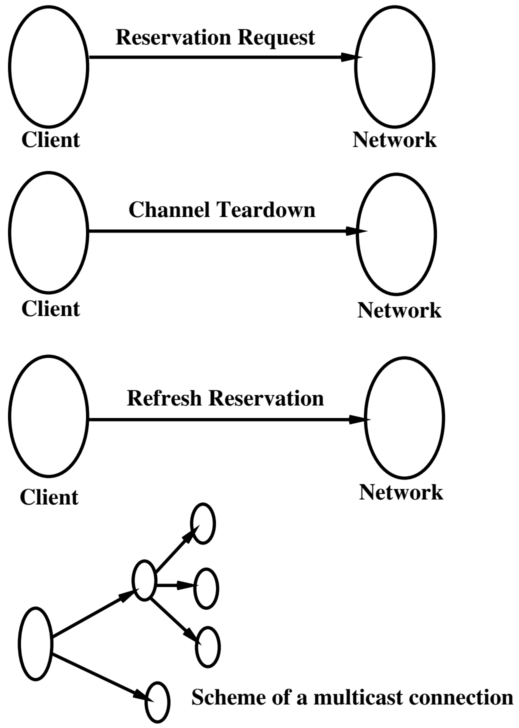

Figure 3.6: Basic resource reservation operations.

RSVP supports heterogeneous receivers efficiently. Not only can the receiver specify its own QoS spec but also the subset of data it is interested in receiving. Furthermore, not all the packets that the sender transmits are delivered to all receivers, but only those the stream specified by the receiver with its QoS requirements.

3.4.5 Critics to RSVP

RSVP presents a high overhead due to:
a. The soft-state resource reservations scheme. Reservation must be refreshed to avoid channels be torn down. The mechanism is not straight forward: there is a tradeoff between overhead one is willing to tolerate and RSVP's responsiveness in adapting to dynamic route changes.

b. New resources must be reserved when a route changes. RSVP's task is to maintain resource reservations over a path or a distribution tree, independent of how the path or tree was created. Now, if the underlying network layer provides a datagram service (e.g. IP) routing is done on a per packet basis, therefore resource reservation overhead may be high. A connection-oriented service would, for example, be better suited for resource reservation.

Another problem is that the service provided may depend on future unpredictable client behaviour. Consider, for example, the case of two receivers placed nearby on the same source tree. A source tree is rooted at the sending node and branches to all receivers. To improve network efficiency, packets are duplicated only at the last node on the source tree common to both routes of the two receiving nodes. Now if resources are reserved only by one of the two receivers (say r1). Now if receiver1 (r1) were to switch rapidly to another sender, it would not longer refresh the resource reservation, and this deprives r2 (and other receivers) of the service. To avoid this dis-service, separate reservations must be maintained or some kind of resource reservation overlapping must be devised.

Finally, since RSVP is designed to run on any routing protocol it is not an optimal solution. Furthermore, resources may be wasted since resource reservation in worst case is blocking, i.e., idle but reserved resources are not available to other connections.

3.5 Virtual Clock: A New Traffic Control Algorithm for Packet Switching Networks

Virtual Clock is a rate-based traffic control algorithm for high-speed networks. It is suited for traffic with relaxed error recovery requirements (i.e. loss tolerant video and/or audio traffic).

Reliable data transfer on low speed networks is better served by a window based flow control mechanism.

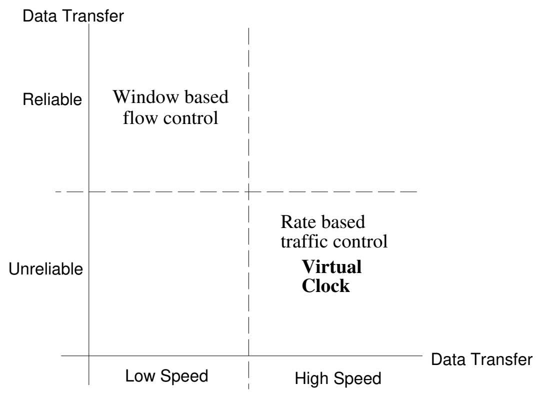

Figure 3.7: Virtual Clock algorithm for high-speed and non reliable data transfer.

3.5.1 Overview

Virtual Clock (VC) is a traffic control algorithm for high-speed packet switching networks. It is a *transmission* control algorithm because it determines the transmission order of packets. It may also be used as an admission control algorithm, i.e., used to decide which packets to admit on the network, which packets to reject of a fast running flow. VC can be used to control the average transmission rate of flows and to enforce that each user's resource usage is in accord with the reservation that it specified at channel setup time. Basically, Virtual Clock considers the traffic control problem as a case of resource scheduling. Here resources are switch processing power and channel bandwidth. VC claims to achieve guaranteed performance through resource reservation and rate based traffic control. However, VC does not guarantee a no packet drop service. In fact VC enforces that a flow's resource usage be proportional to the *average* throughput, but if all senders are transmitting at *peak* rate, packets may be lost because of buffer overflow. Virtual Clock, however, protects flow performance from malicious senders, i.e. senders that are transmitting at more than the specified rate.

3.5.2 Virtual Clock algorithm

The VC algorithm determines the packet transmission order and transmission time without changing the statistical sharing features of packet switching networks. VC algorithm transforms a statistical data flow on the real-time axis into a regular uniform data flow on a *virtual time* axis (see figure 3.8) and moves the transmission times. From this parallel derives the name Virtual Clock.

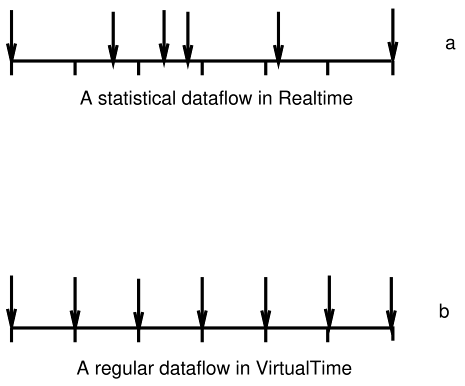

Figure 3.8: Statistical packet arrivals in real time(a) are transformed in uniform packet arrivals in virtual time(b).

Virtual Clock uses two parameters to characterize each data flow:

1. AR_i: average transmission rate of $flow_i(AR_i)$. This is indicated in $flow_i$'s channel set up request.

2. AI_i: average interval with which the algorithm checks the flow's average transmission rate. The average interval is measured on the realtime interval.

Each data flow i is assigned to a Virtual Clock, VC_i. This clock ticks at every packet arrival from the flow of a quantity

$$V_{tick_i} = \frac{1}{AR_i} \tag{3.8}$$

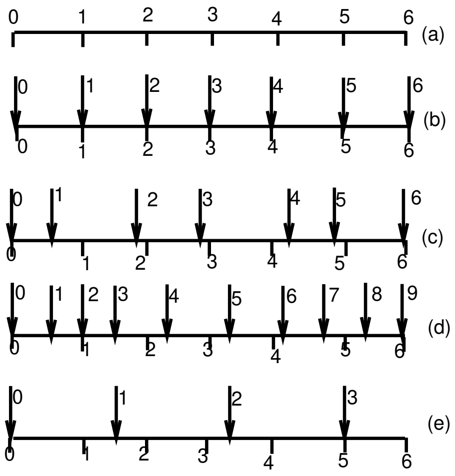

Figure 3.9: Comparison real time and virtual time for AI=6.

called mean inter-packet gap.

An implementation outline of VC follows:

1. Compute V_{tick_i}: for each flow assign $V_{tick_i} = \frac{1}{AR_i}$.
2. Initialize VC: upon arrival of first first packet from flow i, set VC_i = real time.
3. Update VC: upon arrival each other packet from flow i, advance VC_i by V_{tick_i}.
4. Transmit packets: by order of increasing VC time stamps, transmit packets.
5. Eventually drop packets: if buffer space is insufficient, drop last packet from the queue.

Packets are time stamped with VC time and sent by order of increasing VC time stamps.

Figure 3.9a shows the real time axis and six real-time intervals. Virtual times are placed besides the arrows which represent packet arrivals.

If packet arrival has a uniform distribution of value $\frac{1}{AR_i}$, then real time and virtual time coincide (see figure 3.9 b). If a flow sends packets according to its specified average rate, its VC will be in the *vicinity* of real time (see figure 3.9 c).

If a flow's sending rate is higher, then VC runs faster than realtime clock (d); if the flow's sending rate is lower, then VC runs slower than realtime clock (e).

Packets are time stamped with VC time and sent by order of increasing VC time stamps.

An implementation outline of VC follows:

1. Compute V_{tick_i}: for each flow assign $V_{tick_i} = \frac{1}{AR_i}$.
2. Initialize VC: upon arrival of first packet from flow i, set VC_i = real time.
3. Update VC: upon arrival each other packet from flow i, advance VC_i by V_{tick_i}.
4. Transmit packets: transmit packets in order of increasing VC time stamp.
5. Eventually drop packets: if buffer space is insufficient, drop last packet from the queue.

3.5.3 Virtual Clock as a data flow monitor

The VC algorithm may be used to monitor the sender's data transmission rate. In fact, VC time is driven by packet arrivals, and thus if the sender is transmitting at its claimed average rate AR_i, VC time and real clock time are near. For example, if virtual time is much ahead of real-time, then from the periodic comparison between realtime and virtual time(time driven test), we can send feedback information to the senders, telling it to eventually stop transmitting.

How often should we compare the two times? That is which value assign to AI_i. If AI is too small and a burst of packets arrives, at the end of AI the VC is much ahead of real clock time and thus transmission is stopped. For this reason, bursty traffic is not tolerated and control may be instable. On the contrary, if AI is too large, a sender may saturate resources, because overload is not promptly detected. To avoid the later problem, a *packet arrival driven test* can replace the time driven test: we compare real clock time to virtual clock time after $(AI_i) * (AR_i) = (AIR_i)$ packet arrivals, instead of after every AI_i real clock time.

This enables to quickly detect bursts of packets.

An outline of VC flow monitor algorithm follows:

1. At $flow_i$ setup, compute $AIR_i = AR_i * AI_i$.
2. Upon arrival of each set of AIR_i packets,
- if (VC time > real time) then transmission rate is faster than specified transmission rate; eventually take action.
- if (VC < real time) i.e., specified transmission rate is not surpassed, then resynchronize clocks, $VC_i \leftarrow$ realtime.

VC is driven by packet arrival or realtime, whichever runs faster. From a resource allocation viewpoint, credits cannot be accumulated, thus unused resources are gone. If a flow could accumulate credits while idle and then send packets in bursts, then buffer overflow may occur.

3.5.4 Critics to Virtual Clock

Virtual Clock is a rate based traffic control algorithm that can be used in a packet switching network. If implemented in all sending nodes, together with a channel admission control test, it may be able to provide a congestionless service. Lastly, the VC algorithm does not guarantee bounds on end-to-end delays nor a data loss free service.

3.6 The Tenet Scheme 1: A Realtime Protocol Suite.

The Tenet suite is a connection-oriented internetworking protocol suite based upon resource reservation. The suite incorporates the concept of *real-time channel*, a simplex unicast end-to-end connection with performance guarantees (e.g., guarantees on end-to-end delay and packet loss probability). The suite ensures no packet loss due to buffer overflow. It does not eliminate packet-loss due to transmission errors, which however, are decreasing since the deployment of low-error prone fiber optics. Resources are reserved in the nodes traversed by the real-time channel, and the number of real-time channels the network can accept is limited by channel admission control tests which take place before establishing a new channel[DFZ92].

Contrary to other proposed solutions to the realtime communications problem, the Tenet approach can be easily implemented in a wide spectrum of internetworks. This propriety of scalability is of great importance.

3.6.1 Overview

The Tenet scheme 1 is a realtime communication scheme, i.e., a set of algorithms that implemented in the nodes of a network, is capable of offering a realtime service in the sense of providing performance guarantees (see §1.6.4 for definition real-time service) to its clients. It is based upon a simplex-unicast real-time connection with guaranteed performance.

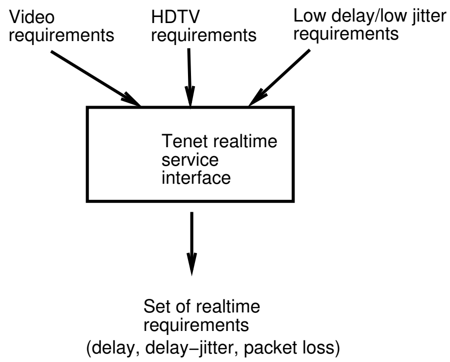

Figure 3.10: The Tenet protocols provide a unique interface for different real-time applications.

As shown in figure 3.10 the Tenet service provides a unique interface for many different realtime application requirements (e.g. video, HDTV, low-delay/bandwidth jitter) and delivers a single set of realtime requirement parameters.

A Tenet suite does not include video protocols nor voice protocols but only the network protocols needed to support the performance bounds (e.g. end-to-end delay, delay-jitter and packet loss) required by real-time applications.

3.6.2 Tenet service model

Tenet Scheme 1 takes into large consideration, the client/service interface, where the service is a network service.

The network/client interface model takes the form of a contract between the client and the network. The client promises to obey certain traffic restrictions and the network undertakes to provide a certain pre-agreed level of service.

This service model implies:

1. *Admission control.* The admission of new real-time channels must be controlled so to ensure that their resource usage does not violate guaranteed realtime services to existing established channel. The admission control criteria is based upon worst-case analysis of scheduling mechanisms and traffic specifications (i.e., all senders transmitting simultaneously at peak rate).

2. *Resource reservation.* Buffer space need be reserved to ensure that packets will not be dropped; likewise switching bandwidth and processing power to ensure an upper bound on packet delivery. The reservation algorithms are based upon worst case analysis of scheduling mechanisms and traffic specifications (i.e., all senders transmitting simultaneously at peak rate).

Both admission control criteria and reservation algorithms are based upon worst case analysis of scheduling mechanisms and the traffic specs. Reservations are not blocking so that free resources can serve best-effort traffic.

3.6.3 Resource reservation

Reserved resources are:
- CPU processing power
- buffer space
- link bandwidth

Reservations may be made for a *guaranteed, statistical* or *best-effort* channel.

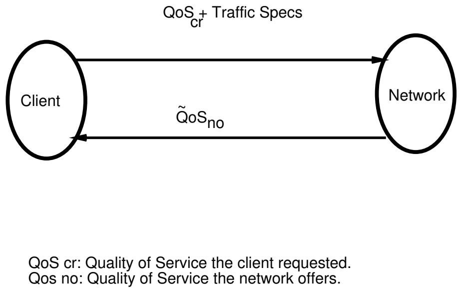

Figure 3.11: Client-Network contract.

As shown in figure 3.11, the client defines the traffic specifications:

• x_{min}: minimum packet interarrival time

• x_{ave}: average packet interarrival time

• S_{max}: maximum packet size

• I: averaging interval for x_{ave}

The clients Quality of Service requirements are:

$\vec{QoS}_{requirements} = (D, Z, W, J, U)$

where D is the end to end delay, Z is the probability that the delay bound is satisfied $Z = Prob\{delay \leq D\}$, W is the probability of packet loss due to buffer overflow, J is the delay jitter (max variation in packet delivery), and U is the probability that delay jitter is satisfied.

The network returns:

$$\vec{QoS}_{offeredbynetwork} = \vec{\tilde{QoS}} = (\tilde{D}, \tilde{Z}, \tilde{W}, \tilde{J}, \tilde{U})$$

• $\tilde{D}$ is the end-to-end delay that the network guarantees the client

• $\tilde{Z}$ is the network guarantee that delay bound is met

• $\tilde{W}$ is the network guarantee on packet loss due to buffer overflow

• $\tilde{J}$ is the network guarantee on jitter

•$\tilde{U}$ is the network guarantee that jitter is met

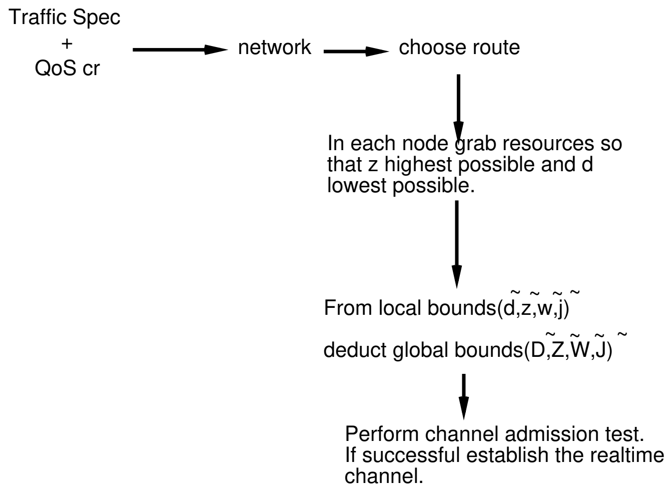

Figure 3.12: Channel establishment phase.

Figure 3.12 describes the channel establishment phase. Channel set up is performed in one round trip through a distributed algorithm.

The client specifies its traffic specs and Quality of Service requirements to the network (QoS client request). A route is chosen for the realtime channel. In each node of the realtime channel's route, the resource manager (RCAP) grabs resources so that z (probability on local delay bound: z=Prob{delay in one node $\leq$ d }) is highest possible ($\simeq$ 1) and d lowest possible. Arrived at the end node, we first calculate the global bounds (e.g., D, Z) from the local bounds (e.g., d and z) and the perform the channel admission test. If this test is successful, the real-time channel is established.

On the forward pass, the algorithm grabs as many resources as possible, i.e., RCAP reserves bandwidth so that local delay is the lowest possible; it also reserves buffer space so that probably of buffer overflow is $\simeq 0$.
At the end of the forward pass, at the destination node, we compare the clients QoS requirements ($\vec{QoS}_{cr}$) to the QoS the network can offer ($\vec{QoS}_{no}$). If

$$\vec{QoS}_{cr} \leq \vec{QoS}_{no} \tag{3.9}$$

[3] that is the channel admission test is passed, then on the reverse pass, the tentative resource reservations made on the forward pass are committed and the channel can be established. To optimize resource usage, on the reverse path, the resource reservations can be relaxed (this facilitates the establishment of new channels), but still as to ensure the clients QoS request is met: if we consider the equality in 3.10 and introduce the relaxed QoS the network offers ($\vec{QoS}_{no-relax}$), we obtain:

$$\vec{QoS}_{cr} = \vec{QoS}_{no-relax} \tag{3.10}$$

.

3.6.4 The protocol suite

The Tenet Suite 1 consists of four protocols as shown in figure 3.13.

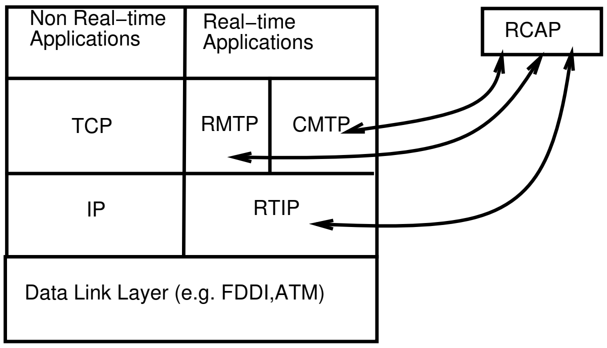

Figure 3.13: Tenet real-time protocol stack.

•Real-time Channel Administration Protocol (RCAP): A resource management protocol that performs channel setup based upon the clients QoS performance requirements, its traffic description, and the availability of network resources.

[3]The test $\vec{QoS}_{cr} \leq \vec{QoS}_{no}$ is called channel admission test. It means:
$prob\{\tilde{D} \leq D\} \geq Z$
$\tilde{W} \leq W$
and we assume no jitter control.

• Real-Time Internet Protocol (RTIP): An internetworking protocol which provides data transfer on a simplex, unreliable, *guaranteed performance* packet service. It schedules packets according to reservations made by RCAP.

• Real-time Message Transport Protocol (RMTP): an end-to-end realtime data transfer protocol which provides an unreliable guaranteed performance message service to the clients. RMTP maintains state at the end points of the channel. RMTP state is used to regulate the rate of packets entering the network, and the fragmentation and reassembly of messages into packets.

• Continuous Media Transport Protocol (CMTP): a transport protocol suited for clients which generate and/or consume data at regular time intervals (i.e., continuous media clients). CMTP provides unreliable, end-to-end, continuous, sequenced, periodic transfer of *stream data units* (STDUs) from one sender to one receiver (simplex) with guaranteed performance. If the client specifies a STDU length of one byte, the the stream becomes a byte stream.

RTIP and RCAP protocols are installed in the network nodes; the transport protocols (RMTP and CMTP) run end to end.

The Tenet protocol suite can coexist with the TCP/IP protocols.

The Tenet approach guarantees performance in a packet-switched network.

Performances guarantees can be:

• deterministic guarantees
• statistical guarantees
• best-effort guarantees

and realtime channels are:

• deterministic: a realtime channel where Z=U=W=1; all packets arrive and arrive on time (guarantees are absolute).

• statistical: a realtime channel where $0 \leq Z * U * W \leq 1$; some packets are lost and/or are late and/or have high jitter (only some guarantees are made).

• best-effort: a realtime channel where Z=W=U=0. No guarantees are made.

Since the resource reservation is done considering a worse case (i.e., all senders transmit at their peak rate), if the client abides to its traffic specifications, then the network QoS is a *priori* guaranteed.

3.6.5 Critics to the Tenet approach

Tenet realtime communications services can be easily implemented in a wide spectrum of internetworking environments. In fact, to adapt Tenet realtime protocol suite to different networks, we need to change only:

1. The admission control test
2. The local bound computations

Tenet resource reservations are non blocking, i.e., reserved resources may be used by non real-time channels.

The Tenet suite1 does not consider multiple destinations[4].

3.7 Summary

Some *realtime* protocols provide predictable performances but cannot ensure that the guarantees not be violated (e.g. Stanford University solution to realtime communications) or predictable only over a short time interval.

Table 3.1 summarizes current solutions to the realtime communications problem. Many schemes support a multicast channel abstraction that is well suited for transmission to a set of receivers.

In RSVP's solution, multicast channels provide different QoS requirements to the receivers (i.e., support heterogeneous receivers);instead, ST-II's multicast channels can support only one QoS requirement equal for all receivers. The Tenet solution (scheme1) can provide guarantees: For example, a deterministic channel can guarantee a no packet loss service and a end-to-end delay bound on a unicast connection.

[4]This is a characteristic of the Tenet scheme 2

[5]co means connection-oriented; cl means connectionless.

Table 3.1: Characteristics of solutions to the Realtime communication problem

Pcol	m/u cast	Het.rec.	Congestion Control	Packet Routing	Service[5]
ST-II	multicast	no	reserves resources but does not guarantee packet loss free service	delagated to other protocols	co
SRP	unicast	-	does not guarantee a packet loss free service	static IP	cl
RSVP	both	yes	Soft state. Worst case is blocking	connectionless IP	cl
VC	-	-	does not guarantee a packet loss free service. does not guarantee upper bounds on end-to-end delay	-	-
Tenet	unicast	-	guarantees no packet loss service. Guarantees upper bounds on end-to-end delays	-source	co

Chapter 4

The Tenet Approach

4.1 Introduction

In this chapter we consider the Tenet approach to real-time communication and the importance of providing a Quality of Service to multimedia traffic. We consider the tests, which, if successfully passed, provide mathematical guarantees on network performance such as end to end delay and packet loss.

4.2 Quality of Service: The Central Issue

Current packet-switched networks suffer congestion. In order to support integrated services on such networks, congestion must be controlled and avoided. There is a much research on congestion control in ATM networks , or more general packet-switched networks[LP91][JHP91].

Let us consider a common classification of network traffic. In figure 4.1 traffic is classified in the cartesian product (delay sensitivity) X (loss sensitivity).

In the first quadrant we place traffic which is delay sensitive and loss sensitive. Such traffic is generated, for example, in the control of a remote Unix process. In the forth quadrant we place traffic which is loss-sensitive but can tolerate delay. For this traffic, a reliable data transfer is sufficient. Traffic generated by interactive video and audio is placed in the second quadrant. This traffic is delay sensitive but

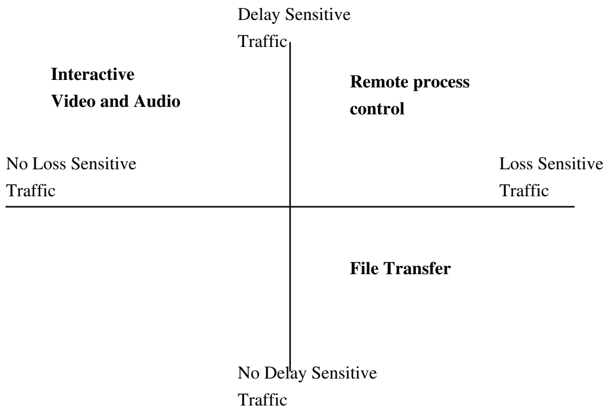

Figure 4.1: Classification of network traffic.

can tolerate packet loss (i.e., the offered service is still acceptable if all packets are not delivered). However, what happens if congestion occurs and many packets are dropped? The client (in this case the client may be a human being) will not be satisfied. He does not have the means to quantize the quality of service offered but might use expressions like "This is not good quality video", or "I cannot understand what the speaker is saying in this audio channel". Similar expressions are qualitative valuations of the service and need be taken into consideration by the service provider (see figure 4.2 4.3 (Source [KV93]).

Increase in bandwidth may be useful, but it is not sufficient (see §1.5.1).

A central issue in integrating all media types on a raw packet switched network (which by itself, provides no guarantee whatsoever) is the Quality of Service the network can guarantee to its clients [DFV92].

Traffic types have different requirements in terms of bandwidth and delay. In figure 4.4 we classify traffic requirements in the cartesian product (Bandwidth requirement) x (Delay requirement).

Traffic generated by e-mail applications does not present strict delay nor high bandwidth requirements (fourth quadrant). Notice instead how low delay is required by interactive video and audio (quadrant two) and remote process control applications (quadrant three).

Figure 4.2: A frame with no packet loss

Figure 4.3: A frame with packet loss

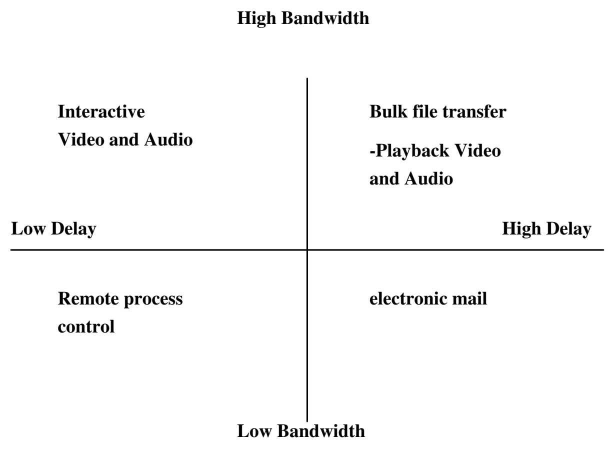

Figure 4.4: Bandwidth and delay requirements.

The use of QoS parameters can assist resource management and help in matching user requirements and traffic characteristics to the network's capabilities. It also provides an abstraction of the network's performance to upper layers: the transport system[1] does not know that the user is opening a voice or video connection; it only sees the QoS parameters. Characteristic QoS parameter values for different traffic streams are shown in table 4.2 (source: [DHS90].)

Table 4.1: Characteristic QoS parameter values for different traffic types.

Traffic Type	MaxDelay(s)	Max Delay Jitter (ms)	Av. Throughput (Mbps)	Acceptable Packet Error Rate
voice	0.25	10	0.064	10e-1
video	0.25	10	100	10e-3
image	1	-	2-10	10e-9

From a service viewpoint, QoS control (and reliability), is more effective in a connection-oriented service rather than in a connectionless.

[1]The transport system transfers bits from the physical layer to the application layer, i.e., it consists of the data-link layer, the network layer and the transport layer.

Often it is difficult to specify the clients QoS requirements because traffic is difficult to characterize. One way to overcome this problem is to associate each traffic type (i.e., video, audio and data) to a *QoS class*, and to specify for each class, the QoS requirements. These may be:

- max cell loss
- max delay
- max number of consecutive cell losses[2]
- min average throughput loss
- max cell loss

This model is used by Lazar for ATM networks [LP91] [JHP91] [Laz92], where QoS is guaranteed, through both buffer management and scheduling algorithms, to classes of calls and not single connections (i.e., virtual circuits). However, class based QoS guarantees are somewhat correlated to virtual call based QoS guarantees, but to different degrees based upon the performance measurement: there is a strong correlation for performance measurements on absolute bounds. For example, if delay is upper bounded for a class of calls, then delay is upper bounded for all calls of that class. Less strong correlation exists for cell loss rate: a low cell-loss rate for a class of calls is directly correlated to a low cell-loss rate for calls of that class. Finally, there is a weak correlation between consecutive cell-losses for aa class of calls and calls of that class.

4.3 The Tests of the Tenet Realtime Service

Quality of service in the Tenet approach is ensured through a sequence of tests to be performed at channel establishment time. The uniqueness of the the Tenet approach is that, if a real-time channel passes the tests, then its network performance is mathematically guaranteed.

In this section, we give a brief description of the tests performed. We assume a network with general topology and store and forward nodes. We also assume

[2] Used in audio QoS to guarantee continuous voice

the real-time channel to be a simplex fixed route connection wheree resources are reserved at connection establishment time and freed during connection disestablishment.

At establishment time the client must :

1. Declare its traffic characteristics, that is the load it places onto the network.
2. Declare its performance requirements, that is the performance bounds it requires from the network.

The client characterizes its traffic through:

•x_{min}: minimum packet interarrival time

•x_{ave}: average packet interarrival time

•S_{max}: maximum packet size

•I: averaging interval for x_{ave}

and performance requirements through performance bounds such as:

•D = end to end delay bound[3]

•$Z = Prob\{\tilde{D} \leq D\}$

•$W = Prob\{$packet loss due to buffer overflow$\}$ (max packet loss rate)

•$J = |\tilde{D}_{max} - D|$ (end to end jitter)

•$U = Prob\{\tilde{J} \leq J\}$

where letters with ˜ notate network measured values of delay ($\tilde{D}$) and jitter ($\tilde{J}$).

The real-time channel is called a *deterministic* channel if the guarantee on end to end delay is a hard guarantee (i.e., valid for each packet) or a *statistical* channel if the bound on D is statistical:

•$Prob\{\tilde{D} \leq D\} \geq Z$

Let's assume a real-time channel i defined by the nodes (1,2,4,6,7) as in figure 4.5.

A node is characterized by $d_{i,j}$ (local delay of a packet in node j), $w_{i,j}$ (local buffer space reserved in node j) and $z_{i,j}$ (local probability of packet loss).

Real-time channel establishment is performed in one round trip: a forward pass

[3]Delay bounds are specified only for packets that reach the destination host.

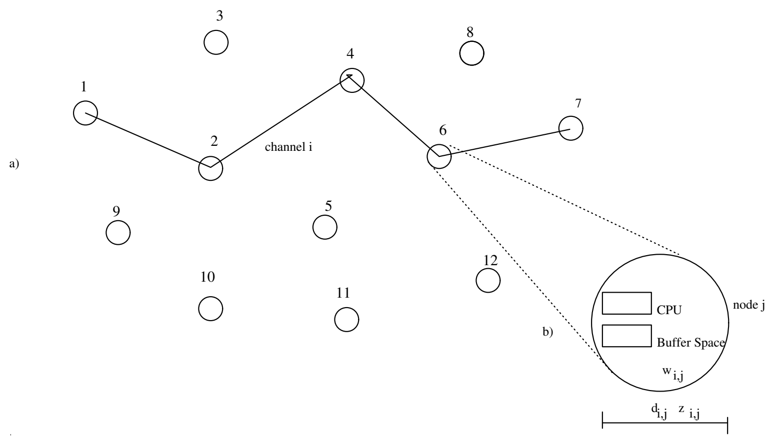

Figure 4.5: A realtime channel a) and resources in a node b)

and a reverse pass.

On the forward pass:

a. In each node resources are grabbed so that

$\bullet\tilde{D}$ lowest possible (packet scheduling resources are reserved) and

$\bullet\tilde{W}$ lowest possible (buffer space resources are reserved).

b. At the destination host, a channel admission test is performed so that *deterministically* or *statistically* $\tilde{W} \leq W$ and Prob$\{\tilde{D} \leq D\} \geq Z$ [FV90b] [FV90a].

On the reverse pass:

Resource reservations are relaxed but as to still respect guarantees (see §3.6.3 for further details on the reservation scheme).

The tests performed in intermediate nodes are called deterministic test, statistical test and delay bound test; those performed at the destination nodes are called Z test, D test, and W test[FV90b][FV90a].

- The deterministic test is a function

D_T :(Node state, new deterministic channel)$\longrightarrow \{yes, no\}$

that given the resource state of a node and a new deterministic channel, is:

a) passed (value=yes) if there is enough *processing power* to accept the new deterministic channel, and

b) not passed (value=no), otherwise.

It is used to verify that enough processing power is available in the node, even in the worst case (i.e., all deterministic channels sending packets into the node at the peak rate).

The condition to be tested is:

$$\sum_j \frac{t_{j,n}}{x_{min.j}} \leq 1 \tag{4.1}$$

where the sum extends to all deterministic channels passing through node n, including the one to be established (channel i).

• The statistical test is a function, that given the resource state of a node and a statistical channel, is passed (value=yes) if the local statistical delay bound is satisfied. That is, if for each old statistical channel passing through node n:

$$Prob\{\tilde{d_{j,n}} \leq d_{j,n}\} \leq z_{j,n} \tag{4.2}$$

This test also calculates $z_{i,n}$, which will be used in the statistical test of a new statistical channel. The *a priori* knowledge of $z_{i,n}$ enables to avoid node saturation[4] which may cause the miss of delay bounds. In fact, over reservation of resources is avoided since we calculate $z_{i,n}$ as the max $Prob\{\tilde{d_{j,n}} \leq d_{j,n}\}$ achievable, compatible with the resources available.

• The delay bound test.

The delay bound test verifies that there is sufficient delay resource for new channels. It also computes the local delay $d_{i,j}$.

[4]Node saturation can occur because of lack of processing power or lack of switching power.

The second cause of missed delay bound is scheduler saturation[5] which, however, is avoided *a priori* by the delay bound test. In fact, the delay bound test calculates $d_{j,n}$ (local delay) so that there is no scheduling saturation.

When a node receives a real-time channel establishment request message, the resource management block (RCAP) performs several tests. In figure 4.6 we show the sequence of these tests performed for the establishment of such channel.

We notice how a real-time channel can be established only if in all nodes on its route have sufficient link bandwidth (or switching power), sufficient buffer space and sufficient processing power. If all these tests are passed, there is no node saturation or scheduler saturation. Finally, in the end host we compare the client's QoS requirements (Z,D,W) to the QoS that the realtime channel can provide $(\tilde{Z}, \tilde{D}, \tilde{W})$. This comparison is performed in the Z, D and W tests.

The sequence of tests performed in intermediate nodes of the real time channel is shown in figure 4.7 where in a) we consider a real-time deterministic channel; in b) we consider a real-time statistical channel; finally, in c) a best-effort channel.

If a deterministic channel is being established (a), a deterministic test is performed; in addition, if a statistical channel passes through the node, then a statistical test is performed; finally, a delay bound test is performed.

If a statistical channel is being established (b), then statistical and delay bound tests are performed.

For best-effort channels (c) no tests are required. In fact, for these channels no resource reservation is required.

Now we consider the end to end tests.

The Z, D and W tests are end-to-end performance indices [6].

• The Z test uses the $\tilde{z_{j,n}}$ calculated for each node n on channel j's route to verify that :

---

[5]the first being node saturation.

[6]Remember that we indicate with the ˜ performance values the network can guarantee.

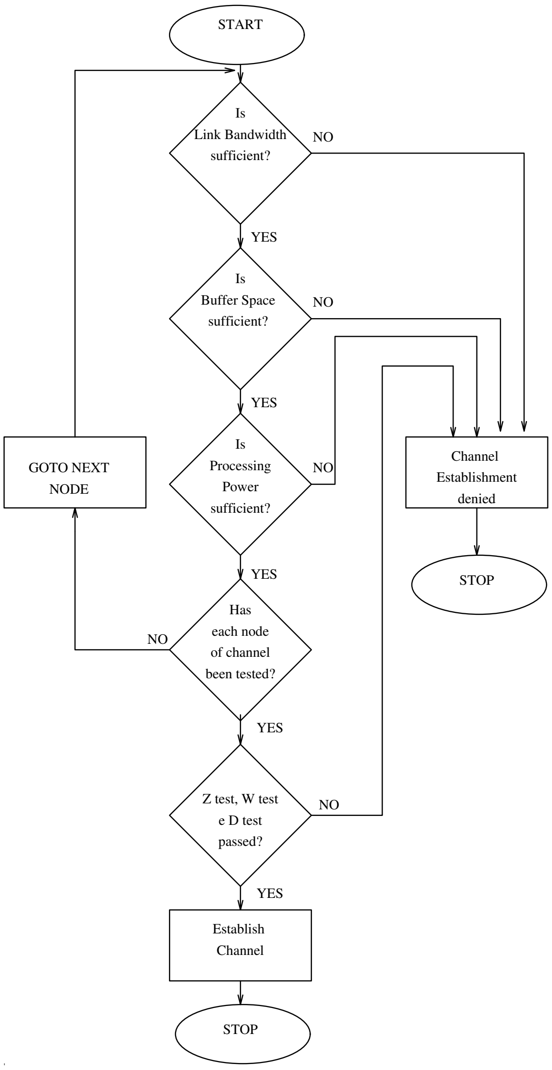

Figure 4.6: Flow chart of simplified tests performed in intermediate nodes and on destination host at channel establishment time.

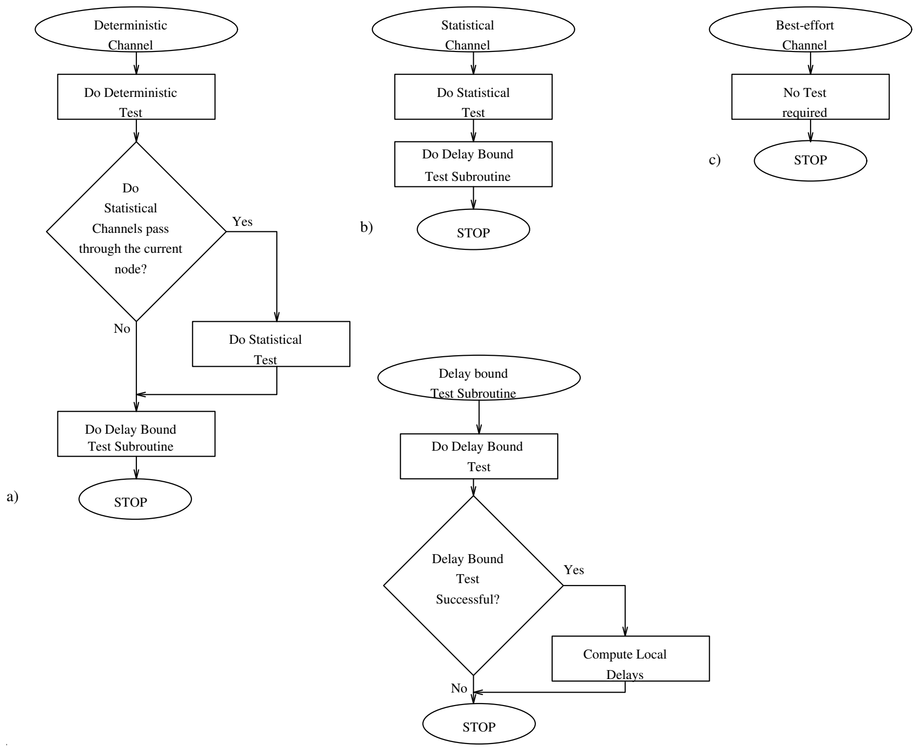

Figure 4.7: Flow chart of tests performed inside the network at real time channel establishment time.

$$Z_j \leq \tilde{Z_j} = \prod_{n=1}^{N} \tilde{z_{j,n}} \tag{4.3}$$

If the 4.3 is satisfied, then the Z test is passed.

• The D test, or end-to-end delay test, uses the $\tilde{d_{j,n}}$ obtained for each node n on channel j's route to verify that:

$$D_j \geq \tilde{D_j} = \sum_{n=1}^{N} \tilde{d_{j,n}} \tag{4.4}$$

If the 4.4 is satisfied, then the D test is passed.

• The W test, or buffer space test, is performed on new real-time channels. It uses the the buffer space tentatively reserved on the forward pass for a real-time channel i, in node n (i.e., $w_{i,n}$) to verify that:

$$W_j \geq \tilde{W_j} = \prod_{n=1}^{N} \tilde{w_{i,n}} \tag{4.5}$$

Chapter 5

Continuous Media Transport Protocol: a Tenet Realtime Transport Protocol.

[1]

5.1 Introduction

In the view of many authors, any effective and efficient congestion control scheme must be capable of performing traffic smoothing functions [DHS90] [WM92] [FAG92] [WM91]. For example, it can be shown that in an ATM network resource utilization improves if burstiness decreases (i.e., less bandwidth need be allocated to guarantee the same QoS if burstiness is low).

The Tenet protocol suite is designed to provide a real-time network service (i.e., a network service with guaranteed performance) to applications. These applications can be divided into two types: those that generate data at arbitrary times, which are called *continuous media(CM)* clients (e.g., videoconferencing, high-quality digital sound, and voice communication); those that generate data at arbitrary times, which

[1]Part of this study was undertaken with Chris Tow as a project in Computer Networks class (CS268) at UC Berkeley, Spring 1993.

are called real time *message oriented* clients.

Since CM clients provide traffic with a certain regularity, they can characterize their future behavior better than real-time message oriented clients. Thus, CM transport services, like the Continuous Media Transport Protocol (CMTP) of the Tenet Real-Time protocol suite, can use this a priori knowledge of future data transmission timing to smooth traffic and thus more efficiently utilize network and end-system resources (e.g., buffers).

The goal of our simulations is to examine the improved performance CMTP provides as compared to that of RMTP.

Why CMTP

Why is an specific transport layer protocol needed for continuous media traffic?

Current transport protocols are not well-suited towards serving realtime or CM clients in a heterogeneous network; the introduction of a specific transport protocol for CM clients in the Tenet real-time protocol suite is an attempt to better service continuous media service requirements.

Transport layer service requirements can be supported through two different approaches.

First, offer a general transport layer service and provide options on this service to adapt to client's service requirements. This is the approach taken by TCP which implements a flow control mechanism based upon a sliding window. The width of the sliding window is small for slow transfers, and increases for faster transfers.

Secondly, split the transport layer vertically into two distinct parts and implement a specific protocol for CM clients and a specific protocol for message oriented clients.

There is a general agreement in literature that transport protocols be tailored to meet the various transport service requirements of end users. Thus, the second approach has been chosen in the Tenet suite, where two transport layer protocols coexist:

1) RMTP (Realtime Message Transport Protocol): designed for message oriented clients which produce data at irregular intervals.

Typical applications which require a real-time message transport protocol are:

• Urgent messages

  • a) distributed applications

  • b) process control applications

• Mail service with guaranteed delivery latency

• Applications for which an irregular data transfer is sufficient (transmission is "message driven")

2) CMTP (Continuous Media Transport Protocol): designed for CM clients which produce/consume digital data isochronously; the service is based upon a data stream communication abstraction. Typical applications which require a continuous media application are:

• Videoconferencing

• Voice communication

• Transmission high-quality digital sound

• Applications which require a regular data transfer (transmission "message driven")

Furthermore, CMTP provides the following functionalities:

• Implements Rate Control

• Enables Self-Timed data transmission

• Provides a Data Stream Abstraction

This resluts in:

• Improvement of buffer usage in the network nodes

• Reduction of interactions between sending/client receiving client

The following two examples describe how the stream representation used in CMTP facilitates the support of client performance requirements.

If a client wants to freeze a video frame, CMTP will stop the stream. If a client requires slow motion, a slower stream is set up. The stream abstraction is more natural for CM clients and works better for the two examples listed above. Without a stream, if the client wanted to freeze a video frame, it would stop sending packets, but the connection would still be there even though it is not being utilized. With the stream abstraction, clients have a better control over use of its resources: in general, one stream is associated with each end-user service requirement (full motion video, slow motion video, freeze frame). Since the user does not know anything about the Quality of Service connection parameters, but sees a high-level transport connection, the stream abstraction hides these details from the user. By having the stream abstraction at an upper layer (i.e, the transport layer), we can easily translate the CM client requirements into QoS parameters (see fig 5.1).

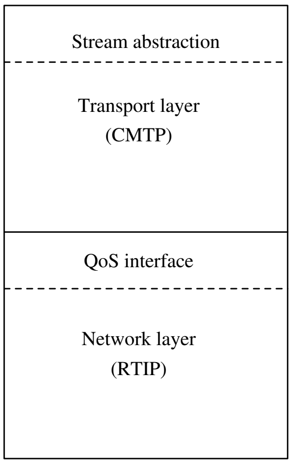

Figure 5.1: Stream abstraction provided by the transport layer and QoS interface provided by the network layer.

From this viewpoint we can restate why CMTP is necessary for CM clients: if we can tailor the transport service to the client's service requirements (and by making CM streams visible at the transport service) then we can use network resources allocated to the transport service connection more efficiently.

5.2 Description of Service and Protocol

The basic components and interactions of the CMTP implementation are illustrated in figure 5.2

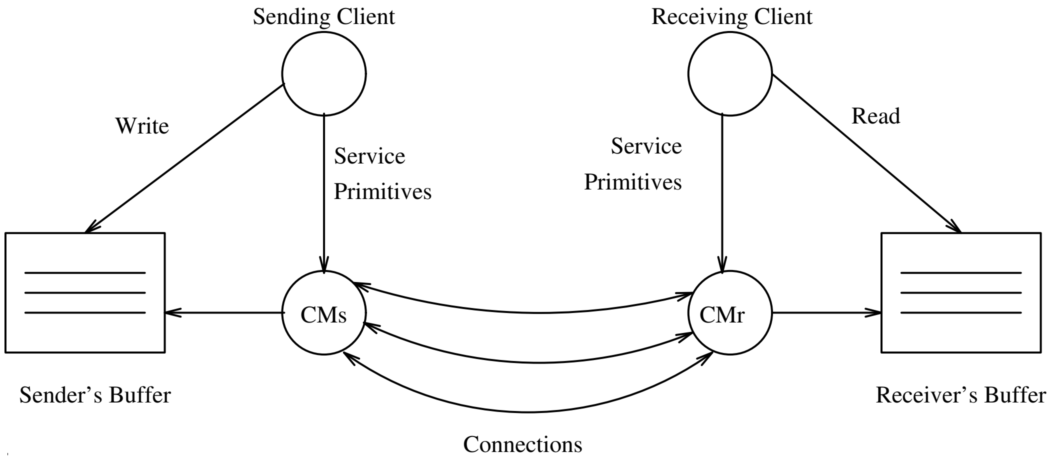

Figure 5.2: Basic interactions of CMTP.

On the left side we have the sending host. It consists of a sending client, a CMTP entity (CMs) and a buffer shared between the two.

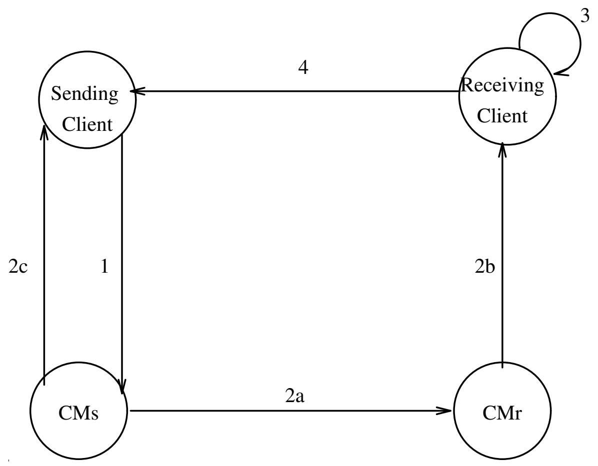

Figure 5.3: Set up Phase of CMTP.

Similarly, on the right side we have the receiving host. The service CMTP offers is divided in a set-up phase and a data transmission phase.

Set up requests are handled on behalf of CMTP by RCAP, which reserves resources at the sending and receiving hosts and establishes a network level real time connection via RTIP.

The main steps of the set up phase are shown in figure 5.3:

1. The sending client (Cs) requests a continuous media (CM) connection from the CMTP entity on the sending side (CMs), specifying its traffic characteristics and the QoS requirements.

2. If the CMTP entity is willing to handle the request, it passes the request to the CMTP entity (CMr) on the receiving side (2a) and thus up to the receiving client (2b). If not, CMs informs Cs that the request was denied (2c).

3. If the receiving client accepts the request, then the connection is established; otherwise, the connection is refused.

4.The sending client is informed of the receiving client's decision.

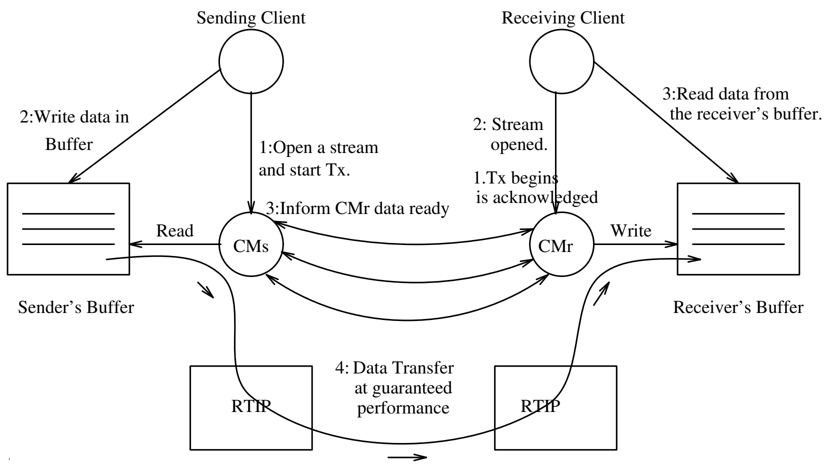

Figure 5.4: Data Transmission Phase of CMTP.

The main steps of the data transmission phase are shown in figure 5.4:

At the sending side.

1. The sending client informs the transport service (CMs) that transmission begins by opening a stream.

2. The sending client writes data into the buffer it shares with CMs.

3. CMs informs CMr that data is being stored in the receiver's buffer. Data transfer occurs through RTIP.

4. Once per period CMs copies the data from the sender's buffer to the receiver's buffer, satisfying the performance requirements specified during set up phase.

At the receiving side.

1. CMr receives an Open PDU packet from its peer entity on the sending side (CMs).

2. CMr informs Cr that the channel has been opened.

3. The receiving client reads data from the receiver's buffer.

Finally, figure 5.5 shows the main steps of the stream closure of CMTP.

1. The sending client informs CMs that it wants to close a stream.

2. CMs empties the sender's buffer (2a) and informs CMr that the stream is being closed by sending a Close PDU packet (2b).

3. CMr informs the receiving client that the stream has been closed.

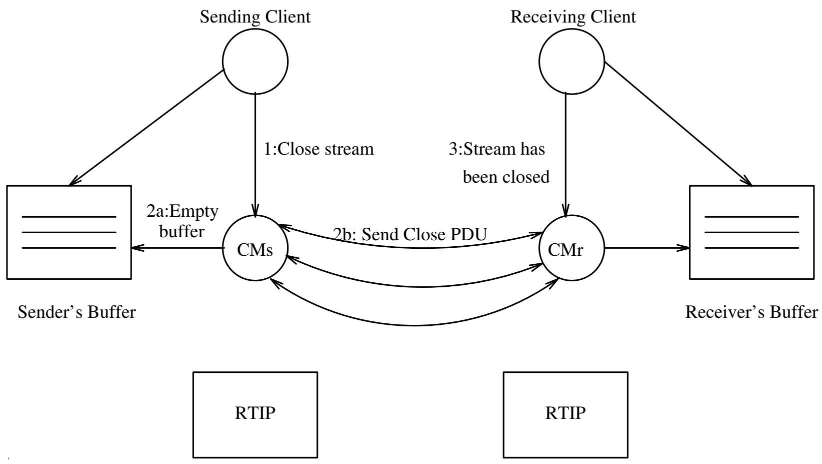

Figure 5.5: Stream Closure of CMTP

5.3 Experiments and Results

We have implemented a simplified model of CMTP on the network simulator Galileo for comparison with RMTP.

Assumptions for the model

In developing our model of CMTP, we chose the following traffic characteristics for traffic generated by CMTP:
- $x_{min} = 2.0ms$
- $x_{ave} = 2.5ms$
- $I = 50ms$[2]

These parameters control the rate at which CMTP entities send data. As long as the traffic that CMTP sends to RTIP behaves according to these parameters, RTIP will guarantee CMTP's performance requirements.

Furthermore, we made the simplifying assumptions:

1. There are always buffered packets ready to be sent.
2. All packets are of the same size.
3. There is no delay jitter control.

Because we assume no delay jitter, and because CMTP is a deterministic time-driven protocol (because of the first assumption), we can consider RTIP to be a work-conserving queue (i.e., no waiting packets are left unserved).

CMTP is deterministic because it sends packets according to the sliding window rate control scheme described in the next section. The assumptions for x_{min}, x_{ave} and I are based on typical values. From these values, CMTP implements its rate control scheme.

[2] An interesting simulation would be simulate how the value of I influences performance.

5.4 Sliding Window Rate Control Scheme

A sliding window is used to *smooth* the traffic sent by the CM client. We introduce the following relation:

$$I = n * T \tag{5.1}$$

where I is the averaging interval defined previously and the width of the sliding window, T is the frame time, and n is the number of frames per window. If the window begins at t=0, the window interval can be represented as:

$$I = I_0 = [0, n * T] \tag{5.2}$$

At time t = T (i.e., the next window) can be represented as:

$$I = I_T = [T, (n+1)T] \tag{5.3}$$

that is, a translation of width T of the original window.

The condition on x_{ave} implies an upper bound on the number of packets that can be sent in a window (i.e., $\frac{I}{x_{ave}}$). The condition on x_{min} implies an upper bound on the number of packets that can be sent in T (i.e., $\frac{T}{x_{min}}$). Considering CMTP's sliding window rate control scheme, the maximum number of packets that we can send in the last frame of an interval is determined by the conditions on x_{min} and x_{ave}:

$$MaxPackets = min\{\frac{T}{x_{min}}, \frac{I}{x_{ave}} - NP_{n-1}\} \tag{5.4}$$

where NP_{n-1} is the number of packets sent in the previous (n-1) frames.

This equation, for the numeric values of T=10ms, I=50ms, n=5, x_{min}=2.0ms, x_{ave}=2.5ms and a packet arrival distribution as shown in figure 5.6 becomes:

$$MaxPackets = min\{5, 3\} = 3 \tag{5.5}$$

RMTP does not implement a sliding window rate control scheme. Therefore, it will send packets at intervals of x_{min} until the x_{ave} condition does not hold (i.e., when $\frac{I}{x_{ave}}$ packets have been sent) and then it will wait until the next interval (I)

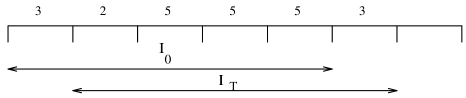

Figure 5.6: An example of how many packets arrive per time frame T to RTIP using the sliding window rate control scheme.

before sending further packets (see figure 5.7). This behaviour affects the number of buffers utilized in RTIP.

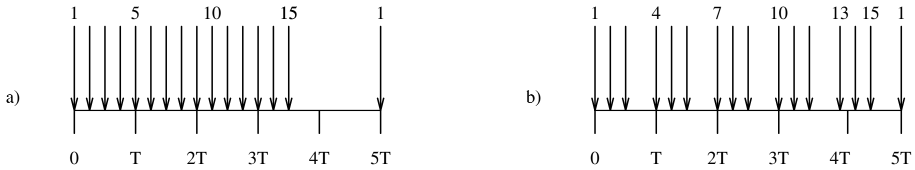

Figure 5.7: Packet interarrival distribution over I (I=5T) for RMTP (a) and CMTP (b).

5.5 Simulation Results

Using our simplified model of CMTP for this study, we show that CMTP utilizes buffers more efficiently than RMTP because it smooths traffic. We ran many simulations comparing the number of buffers used when a particular number of CMTP or RMTP clients were running. In this preliminary study, we show the results produced by four CMTP clients versus four RMTP clients sending packets at the peak rate (worst case). The number of buffers used is shown in figure 5.8.

As the figure shows, CMTP uses buffers more efficiently than RMTP because of its sliding window rate control scheme. When using CMTP, the maximum number of buffers used is three, while for RMTP the maximum number is over twenty. Since CMTP smooths traffic, the number of buffers used oscillates between zero and three. In contrast, RMTP does not smooth traffic , so the number of buffers used increases

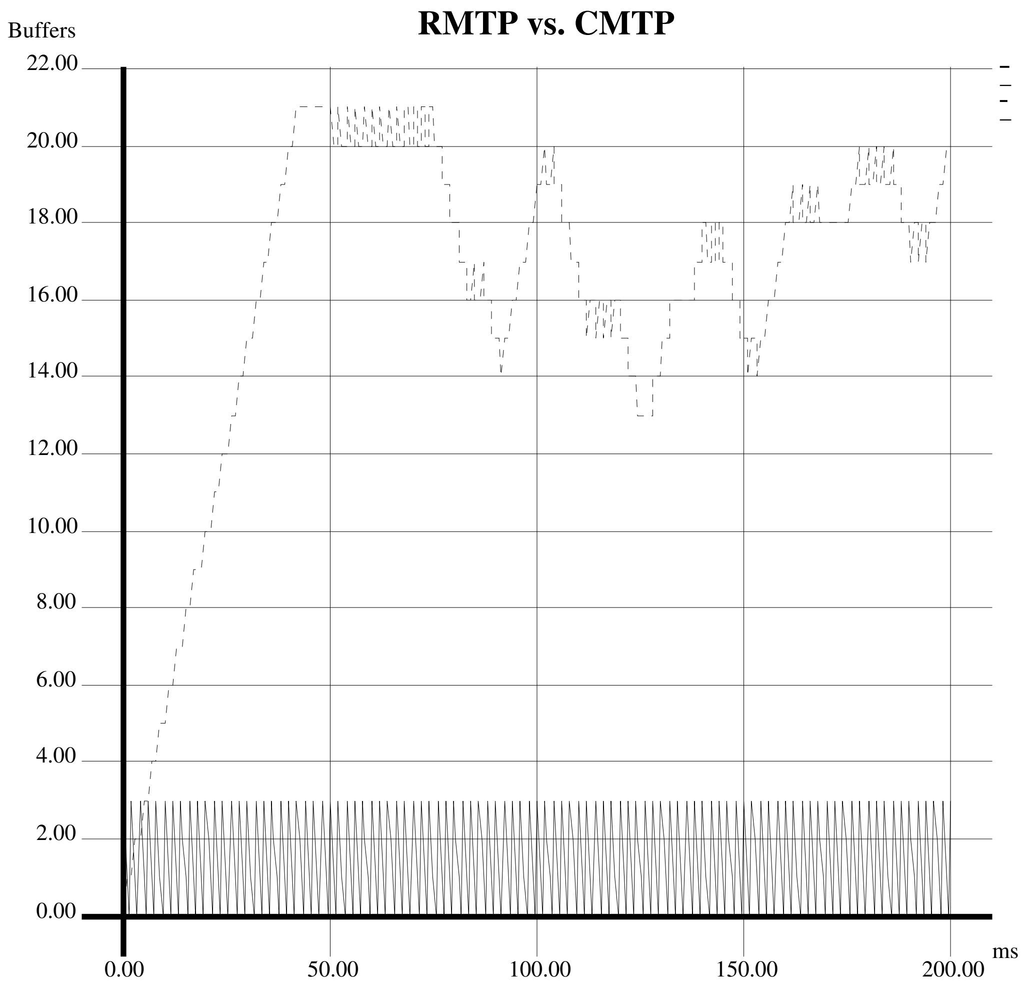

Figure 5.8: Buffers utilized for four RMTP clients vs. four CMTP clients.

and then oscillates. The initial upward sloping portion of the graph (for $t < 50$) occurs because the sources are sending packets at an interarrival rate of x_{min}.

This continues until the x_{ave} condition is no longer satisfied (around t=40). In fact, RMTP sends packets at 2 ms intervals. Thus in I (of width 50 ms) we can send 20 packets ($\frac{I}{x_{ave}}$). If the first packet is sent at t=0, and further packets at intervals of 2ms, the last packet (number 20) is sent at t=40 ms. The subsequent plateau means that RTIP is sending out packets at the same rate that they are arriving ($40 < t < 70$). The successive oscillations are due to a particular source that either: stops sending packets (x_{ave} condition is no longer satisfied) or starts sending packets again (new interval for I).

Initially, we thought that the number of connections could increase when using CMTP because it smooths traffic. However, we realize that even though CMTP sends traffic at a more regular rate than RMTP, the total amount of data does not vary. Thus, the maximum number of connections possible would be the same regardless of whether CMTP or RMTP were used. The performance gain realized would be that fewer buffers would be needed by CMTP.

5.6 Conclusions

The goal of the simulation was to examine how CMTP can improve network resource utilization (buffers) when considering CM clients. Because of the regularity of data produced by CM clients, CMTP was able to *smooth* it using a priori knowledge about the timing of future data transmission.

We described the sliding window rate control scheme used by CMTP to smooth traffic. Since RMTP is unable to smooth traffic, we showed that it caused RTIP to use more buffers than did CMTP. These preliminary results provide justification for having a separate transport layer protocol for CM clients and the continued implementation of CMTP in the internetwork. RMTP would still be used for message oriented clients.

Chapter 6

Future Trends

It is always difficult to predict what the state of any rapid technology will be even in only one decade. This is even more the case of a young, but very rapidly growing industry such as the computer and communications industry. Electronic computers have been around for 40 years and there are clear growth trends that fit computer architecture development over the last 10 years[HP90a]. This is analyzed in the first section.

Trends in network services and internetworking, analyzed in section 2, are closely related if not a direct outcome of this growth.

6.1 Trends in Computer Architectures

Today computer architectures are a bottleneck for providing multimedia on computer networks.

The Amdahl/Case Rule states:

> *A balanced computer system needs about one megabyte of main memory capacity and one megabyte per second of I/O bandwidth per each mips of CPU performance.*[HP90b]

In table 6.1 we consider the growth trends of computer resources (CPU, main memory, disk memory and network bandwidth) where the values are to be considered as orders of magnitude. We consider values relative to 1980 and 1990 from which we calculate the measured annual growth rate. We then extrapolate with the calculated growth rate to the year 2000 (figure 6.1, 5th column). In order to offer a term of comparison to these values, we extrapolate the values of 1990 according to the theoretical growth rate shown in column 4.

In conclusion, we can state that in the year 2000, we will have machines with GIP [1] CPUs, 10's of Gb RAM, 10's of Gb disk (such machine is called a 3 G machine) and gigabit per second networks.

Table 6.1: Trends in Computer Architectures.

Hw Resource	1980	1990	Annual growth rate (measured)	Annual growth rate (theoretical)	2000 (measured)[20]	2000 (theoretical)
CPU Speed(MIPs)	5	150	47%	50%	4500 ($\cong 4Gbps$)	10000
Main Memory(Mb)	1	128	110%	60%	16000 ($\cong 10Gbps$)	14000
Disk Memory(Mb)	1	1000	200%	25%	1000000	10000 ($\cong 10Gbps$)
Network bw(Mbps)	10	1000	100%	50[19]%	100000	57000

Table 6.2 shows the trends of the ratio number users per microprocessor; cost of a microprocessor; and the cost of microprocessor dedicated to a single user.

[1] 1GIP=10^9 instructions per second

[19] [Fra91] shows that transmission speed has increased by 50 perccent per year since 1980.

[20] Calculated considering theoretical value (source: [HP90a]).

Table 6.2: Trends in CPU resource dedicated to a single user.

	1950	1960	1970	1980[21]	1990[22]	2000
users per μp	1000	100	10	1	.1[23]	0.01
cost per μp	\$10M	\$1M	\$100k	\$10k	\$1k	\$100
μp cost associated to single user	\$10k	\$10k	\$10k	\$10k	\$10k	\$10k

Table 6.3: Network technology development over the last 30 years

Year	Technology	Bandwidth(Mbps)
mid70	Ethernet	10
mid80	FDDI	100
mid90	ATM	150-622

Notice how microprocessor cost reduces by a factor of ten each decade and the cost associated to the single user has remained constant.

Computer network bandwidth also seem to follow the 10 times increase in 10 years trend. Network technology and the bandwidth it can provide is shown in table 6.3.

6.2 Trends in Services and in Internetworking

In this section we consider trends in network services and in internetworking. Because of the cost of true gigabit access, it has been estimated that gigabit services will not be available to commercial customers until 2005 [RS92]. However, future network technology need support a wide variety range of applications, with many

[21] 1980 saw the advent of the personal computer.

[22] The 1990s will see the deployment of application oriented microprocessors:one application per microprocessor.

[23] That is, 10 μprocessors per user.

different traffic characteristics, on a wide range of different networks, and through multiple networks (internetworking).

In the past, the design and operation of computers has focused on the task of maximizing processing power once the data is in memory, and not the larger problem of maximizing the performance of networked applications.

The deployment of high speed networks and multimedia computers are correlated: widespread use of high speed networks will depend in part on the degree to which computer companies design their workstations to be fully intergrated into a high speed network.

Studies [DCS89] have shown that if TCP were properly implemented, it could support very high-speeds. Speed up can be obtained by squeezing protocol layers in the implementation of the suite itself [CT90] or by parallelizing communication protocols. In the latter case, performance gains of factor of a few units are common:

Watson has implemented TCP on the parallel Thinking Machine and obtained data rate of 32 Mbps; TCP/IP on a Sparc10 with 4CPUs can reach 100Mbps throughput.

Other research has been undertaken to study the feasibility of parallelization of the Tenet protocol suite [SP93]. This study shows that parallelizing at the packet level seems to be the best compromise but requires major changes in protocol implementation.

The study also shows that critical performance factors are buffer copying and management rather than actual protocol processing. Other studies [ZT91] [DCS89] have come to the same conclusion.
What will the communication scenario look like in a few years? Two trends seem to be forming in the services clients are requesting from the network: the need for multicast connections and the increase of *intelligence* in the network (i.e., supply services other than simple data transfer).

Table 6.4 shows the new services computer networks can support to interactive multimedia applications (source: [SV90])

Trends also show that networks (telecommunication and computer) will need to be more intelligent to provide more useful information and new services to the

Table 6.4: Examples of new services requested from a network.

Type	Description	Data Flow	Accessibility	Event Scheduling
Computer support to collaborative work	All participants actively send/ receive data	N $\rightarrow$ N	Controlled	Planned
Telepresence	Many senders, one receiver	N $\rightarrow$ 1	Controlled	Unplanned
Virtual Cafè	All participants actively send/ receive data	N $\rightarrow$ N	Uncontrolled	Unplanned
Broadcasting Services	One sender, many receivers	1 $\rightarrow$ N	Controlled	Planned
Distributed Computing	All participants actively send/ receive data	N $\rightarrow$ N	Controlled	Unplanned

end user. Only 10% of the cost of a connection will be for the delivery of the information (i.e., the carrier cost) and the remaining 90% will be for both the service provided (e.g., the network service provided for video and audio transmission) and the information itself (for example, information stored in databases).

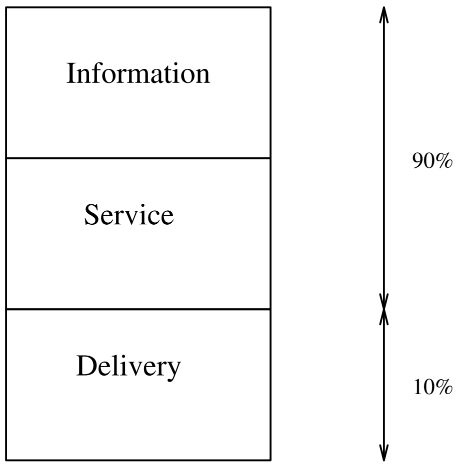

Figure 6.1: How the cost of a communication connection is repartitioned

Finally, figure 6.2[Dec92] shows the bandwidth/intelligence requirement of new services that will fulfill the promise:

to communicate
with anyone
at any time
anywhere
through any information format.

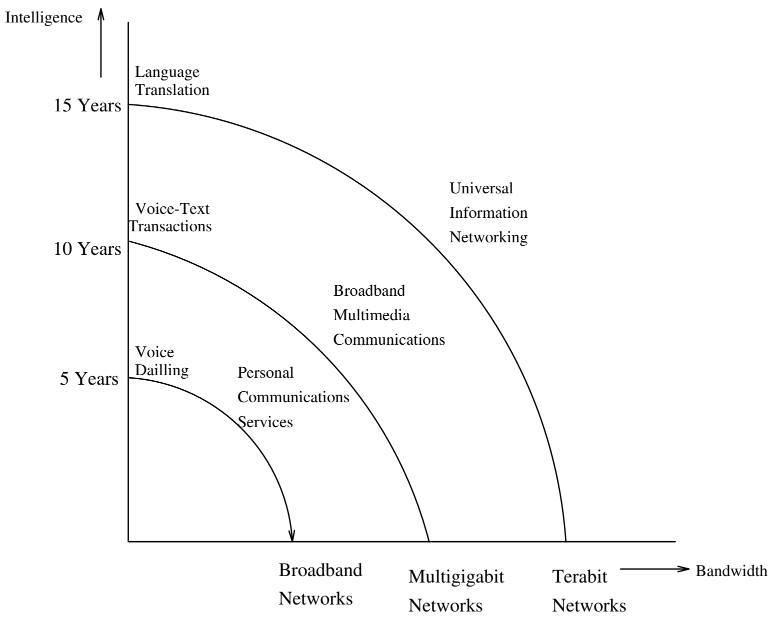

Figure 6.2: Services moving into the information age

Bibliography

[AD89] H. Ahmadi and W. Denzel. A survey of modern high-performance switching techniques. *IEEE J. Selected Areas Comm.*, September 1989. pp 1091-1103.

[Cac89] R. Caceres. Separating abstraction from implementation in a communication network design. *Computer Science Division University, University of California, Berkeley, California.*, 1989.

[CCI80] CCITT. *draft recommendation I.364 on Connectionless Broadband Data Service.* Genève, Switzerland, 1980.

[Coo91] S. Cooper. Gigaplexing fddi and fddi ii bitstreams in local environments. *SPIE Vol. 1577 High-Speed Fiber Networks and Channels*, 1991.

[CS92] J.D. Cavanaugh and T.J. Salo. Internetworking with atm wans. *available via anonymous ftp to ftp.magic.net*, December 1992.

[CSS92] D. Clark and L. Zhang S. Shenker. Supporting real-time applications in an integrated services packet network: Architecture and mechanism. Technical report, LCS, MIT and Xerox PARC, 1992.

[CT90] D. Clark and D. Tennenhouse. Architectural considerations for a new generation of protocols. *ACM*, 1990. 200-208.

[CVN87] R. Herrtwich C. Vogt and R. Nagarajan. Heirat: The heidelberg resource and administration technique. design philosophy and goals. technical report no. 43.9213. Technical report, European Networking Center, Heidelberg Germany, IBM DEUTSCHLAND GMBH, 1987.

[DCS89] J. Romkey D. Clark, V. JAcobson and H. Salwen. An analysis of tcp processing overhead. *IEEE Communications Magazine*, June 1989. pp 23–29.

[Dec92] M. Decina. Network evolution towards universal personal communications. *IEEE NOMS 1992*, 2, 1992.

[DFV92] J. Ramaekers D. Ferrari and G. Ventre. Client-network interactions in quality of service communication environments. Technical report, International Computer Science Institute, Berkeley, California, 1992.

[DFZ92] A. Banerjea D. Ferrari and H. Zhang. Network support for multimedia - a discussion of the tenet approach. Technical Report TR-92-072, International Computer Science Institute, Berkeley, California, October 1992.

[DHS90] M.G. Salmong D.B. Hehmann and H.J. Strittgen. Transport services for multimedia applications on broadband networks. *Computer Communications*, 13, 1990.

[dP91] M. de Prycker. *Asynchronous Transfer Mode. Solution for Broadband ISDN*. ELLIS HORWOOD, 1991.

[Ebr92] Z. Ebrahim. A brief tutorial on atm-draft. *available via anonymous ftp to datanet.tele.fi*, March 1992.

[Egl90] P. Egloff. *Multimedia Applications in a Broadband Environment. Framework for Integrated Services.* Springer-Verlag, 1990. pp85–104.

[FAG92] D. Ferrari and B. Wolfinger A. Gupta, M. Moran. A continuous media communication service and its implementation. *Proc. GLOBECOM'92, Orlando, Florida, USA*, December 1992.

[Fer90] D. Ferrari. Client requirements for real-time communication services. *IEEE Communications Magazine*, 28(11), November 1990. pp. 65-72.

[FL91] H.J. Fowler and W.E. Leland. Local area network traffic characteristics, with implications for broadband network congestion management. *IEEE J. Selected Areas Comm.*, September 1991. pp 1139–1149.

[Fra91] A. G. Fraser. Designing a public data network. *IEEE Communications Magazine*, October 1991. pp 31-35.

[FV90a] D. Ferrari and D. Verma. Buffer space allocation for real-time channels in a packet-switching network. Technical report, International Computer Science Institute, Berkeley , California, 1990.

[FV90b] D. Ferrari and D. C. Verma. A scheme for real-time channel establishment in wide-area networks. *IEEE Journal on Selected Areas in Communications.*, 8(3), April 1990. pp 368–379.

[GP90] M. Gagnaire and B. Ponsard. Evolution of the dqdb hybrid multiplexing for an integrated service packetized traffic. *SPIE Vol. 1364 FDDI, Campis-Wide, and Metropolitan Area Networks*, 1990.

[HP90a] J.L. Hennessy and D.A. Patterson. *Computer Architecture. A Quantitative Approach.* Morgan Kaufmann Publishers, Inc., 1990.

[HP90b] J.L. Hennessy and D.A. Patterson. *Computer Architecture. A Quantitative Approach. Rule of thumbs, pp17-18.* Morgan Kaufmann Publishers, Inc., 1990.

[HS92] I. W. Habib and T. N. Saadawi. Multimedia traffic characteristics in broadband networks. *IEEE Communication Magazine*, July 1992.

[Ich91] T. Ichihashi. Evaluation of fddi-ii cycle synchronization mechanisms. *High-Speed Networking, III-IFIP*, 1991.

[Jac88] Van Jacobson. Congestion avoidance and control. *Proceedings from ACM SIGCOMM '88 Symp.*, September 1988. pp. 314-329.

[Jai90] R. Jain. Congestion control in computer networks: Issues and trends. *IEEE Network Magazine*, May 1990. pp. 24-30.

[JHP91] A.A. Lazar J.M. Hyman and G. Pacifici. Real-time scheduling with quality of service constraints. *IEEE Journal on Selected Areas in Communications*, 9(7), September 1991.

[Kle92] L. Kleinrock. The latency/bandwidth tradeoff in gigabit networks. *IEEE Communications Magazine*, April 1992.

[KP92] J. Kay and J. Pasquale. A performance analysis of tcp/ip and udp/ip networking software for the decstation 5000. Technical report, SEQUOIA 2000, 92/18, 1992.

[Kun92] H.T. Kung. Gigabit local area networks: A systems perspective. *IEEE Communications Magazine*, April 1992. pp 79-89.

[KV93] E. Knightly and G. Ventre. Galileo: A tool for simulation and analysis of real-time networks. Technical Report TR-93-008, International Computer Science Institute, Berkeley, California, March 1993.

[Laz92] Pacifici et al Lazar. Control and management of enterprises comet research group. activity report:1990-1992. Technical report, Center for Telecommunications Research, Columbia University, New York, New York, 1992.

[LD93] F.O. Hoffmann L. Delgrossi, R.G. Herrtwich. An implementation of st-ii for the heidelberg transport system. technical report no. 43.9303. Technical report, European Networking Center, Heidelberg Germany, IBM ENC, 1993.

[Loh91] J. C. Loh. Atm packet speech simulation using ptolemy. Master's thesis, University of California at Berkeley, Berkeley, CA, 1991. call number: T7.49 1991 L5 823 ENG I.

[LP91] A.A. Lazar and G. Pacifici. Control of resources in broadband networks with quality of service guarantees. *IEEE Communications Magazine*, October 1991.

[LS92] J.B. Lyles and D.C. Swinehart. The latency/bandwidth tradeoff in gigabit networks. *IEEE Communications Magazine*, April 1992.

[LZ93] S. Deering et. al L. Zhang. Rsvp: A new resource reservion protocol. *PRELIMINARY DRAFT*, 1993.

[Pax93] Vern Paxon. Growth trends in wide-area tcp connections. *available via anonymous ftp to ftp.ee.lbl.gov*, May 1993.

[PT90] G. M. Patulkar and J. S. Turner. Towards a framework for high-speed communication in a heterogeneous networking environment. *IEEE Network Magazine*, March 1990.

[RS92] M. Ransom and D. Spears. Applications of public gigabit networks. *IEEE Network*, 6(2), March 1992.

[SP93] Wolfgang Schröder-Preikschat. On the parallization of communication protocols-a tenet/cm5 case study. March 1993.

[SV90] C. Szyperski and G. Ventre. Efficient multicasting for interactive multimedia applications. Technical Report TR-93-017, International Computer Science Institute, Berkeley , California, March 1990.

[WC91] C. Williamson and D. Cheriton. Loss-load curves;support for rate-based congestion control in high-speed datagram networks. *Proc. ACM SIGCOMM '91 Symp., Zurich, Switzerland*, September 1991. 17-28.

[Wid93] R. Widyono. Evaluation of routing algorithms for connection oriented multicast channels. *CS266 Performace Analysis of Computer Systems. Final Report. Dept. EECS, UCBerkeley, California.*, May 1993.

[WM91] B. Wolfgang and M. Moran. A continuous media data transport service for real-time communication in high-speed networks. *Proc. Second Internatinal Workshop on Network and Operating System Support for Digital Audio and Video. Heidelberg, Germany*, November 1991.

[WM92] B. Wolfinger and M. Moran. Design of a continuous media communicaton service and protocol. Technical Report TR-92-019, International Computer Science Institute, Berkeley, California, April 1992.

[ZT91] M. Zitterbart and A. Tantawy. Transport services and protocols for high-speed networks. *SPIE Vol. 1577 High-Speed Fiber Networks and Channels*, 1991.