Faculty of Electrical Engineering, Mathematics and Computer Science
Network Architectures and Services

Measurement Study of Multi-point Videoconferencing

Yong Zhao

(1385917)

Committee members:

Supervisor: Prof.dr.ir. Piet Van Mieghem
Mentor: ir. Yue Lu
Dr.ir. Fernando Kuipers
Member: Prof.dr.ir. Piet Van Mieghem
Dr.ir. Fernando Kuipers
Dr.ir. Ertan Onur
ir. Yue Lu

11th December, 2009

M.Sc. Thesis No: PVM 2009 –061

To my dear parents and grandpa…

Measurement Study of Multi-point Videoconferencing

THESIS

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

In

Computer Engineering

By

Yong Zhao

Network Architectures and Services Group

Faculty of Electrical Engineering, Mathematics and Computer Science

Delft University of Technology

Acknowledgment

My master thesis was carried out at the Network Architectures and Services (NAS) Group, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, where I worked from January 2009 to November 2009. I would like to thanks all the people who gave me their suggestions and help during this period.

First of all, I would like to express my appreciation to my Responsible professor Prof. dr. ir. P. F. A. Van Mieghem, my supervisor Fernando A. Kuipers, and my daily mentor ir. Yue Lu. Without your guide and help, I could not accomplish my thesis. I really enjoy the discussion with you, which can always bring me some new ideas and important suggestions. I also would like to thank ir. Feifei Huo, who gave me some good suggestions about the objective and subjective measurements of video quality.

Secondly, I would like to thank my friends, especially my roommates Lin Li and Kai Lin. When I was in trouble or felt upset, you cheered me up and encouraged me to move on.

Finally, I would like to express my special thanks to my dear parents. Without your support, I could not reach here.

Abstract

From the 80s, with the rapid development of the Internet, videoconferencing made significant progress and played an important role in economics, education and remote healthcare. A lot of applications and models appeared. More and more companies and people get used to communicate through a video conferencing service. Therefore, it is important to measure and analyze their quality and performance for the researchers who want to develop new applications. In this thesis, we made a survey about the existing Desktop video conferencing applications on the Internet, and then picked four typical video conferencing applications as experiment objects. We investigated the mechanisms of the application, their traffic characteristics and their communications structures. Through global experiments, we analyzed the different aspects of Quality of Experience (QoE) for these applications. Audio quality, video quality, interactivity level, audio/video synchronization and worst cases are studied. The measurement results help us to understand the behavior and mechanism of video conferencing, and also the QoE of different applications.

Key words: videoconferencing, survey, mechanism, QoE

Table of Contents

Abbreviation

VC	videoconferencing
C/S	Client–Server
P2P	peer-to-peer
NVP	Network video protocol
PVP	Packet video protocol
ISDN	Integrated Services Digital Network
ITU	International Telecommunications Union
RTCP	Real-time Transport Control Protocol
VTB	Video device timebase
ATB	Audio device timebase
BRI	Basic rate interface
PRI	Primary rate interface
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VQM	Video Quality Metric
BVQM	Batch Video Quality Metric
QoE	Quality of Experience
NAT	Network Address Translation
MOS	Mean Opinion Score
PESQ	Perceptual Evaluation of Speech Quality
PESQ-LQ	Perceptual Evaluation of Speech Quality-Listening Quality
DTMF	Dual-tone multi-frequency

List of Figures

List of Tables

1

Introduction

1.1 Background

Video Conferencing (referred to as VC in the following text) conducts a conference between two or more participants at different sites by using telecommunications to transmit audio and video data. During the conference, documents, computer-displayed information and whiteboards can also be shared among participants.

VC was first introduced as a Picturephone at the 1964 World's Fair in New York. But when it was offered in 1970, the high cost turned out its biggest problem ($160 per month). After several years, some new technologies were applied in VC, including network video protocol (NVP) and packet video protocol (PVP) in 1970s. However, it was still far away to put it into commercial use. Till 1980s, digital telephony transmission networks became possible, such as Integrated Services Digital Network (ISDN), assuring a minimum bit rate (usually 128 kilobits/s) for compressed video and audio transmission. The first dedicated systems started to appear in the market as ISDN networks were expanding throughout the world. VC systems throughout the 1990s rapidly evolved from highly expensive proprietary equipment, software and network requirements to standards based technology that is readily available to the general public at a reasonable cost. Finally, in the 1990s, IP (Internet Protocol) based videoconferencing became possible. More efficient video compression technologies were developed, permitting desktop, or personal computer (PC)-based videoconferencing. Figure 1.1 shows the history of VC development.

Figure 1.1 History of Video Conference

From 2000, with the rapid development of Internet, videoconferencing progressed a lot and played a significant role in economics, education and remote healthcare. According to the data from In-Stat and Wainhouse Research, the Unified Communications markets' global revenues are estimated to grow from $22.6 billion in 2007 to $48.7 billion in 2012. Meanwhile, the global market for videoconferencing endpoints was $1.1 billion in 2007, and will grow to $3.9 billion in 2014, which represents a compound annual growth rate of 18%, according to a report of Frost&Sullivan. According to the forecast, the personal market including both PC-based devices and videophones is poised to experience over 60% growth in units and over 40% growth in revenues and will represent a $300 million dollar market in 2009, despite changes in product mix that will drive prices down sharply.

A lot of new VC applications and models come up. Information transfer has become faster and cheaper, more and more companies and people get used to communicate with video conferencing service.

1.2 Motivation

Increased Internet speeds together with new possibilities for real-time interactive multimedia streaming services have spurred the interest in providing multi-use video conferencing via the Internet. Lots of free multi-user video conferencing applications are readily available and both Server-to-Client and Peer-to-Peer technologies are used. Their traffic characteristics, architectures and the Quality of Experience (QoE) provided by them are not well known. Therefore, investigating their mechanisms, analyzing their system performance, and measuring their Quality of Experience are important objectives for researchers, developers and end users.

In this thesis, we focus on studying the free applications providing Multipoint (>=3) desktop videoconferencing on the Internet. We have surveyed a lot of popular free multi-user videoconferencing applications and especially investigated and measured four representive applications: Mebeam, Qnext, Vsee and Nefsis.

1.3 Related Work

VC can reduce travel costs, improve employee productivity, and enable wider participation in decision making. It can also blur geographic boundaries, help to create a consistent corporate culture, and enhance employees' quality of life. More and more people get used to VC in their daily life.

According to the large global market, VC service is a very hot topic now. Lots of work and researches focus on the network architectures and streaming technologies. However, few measurement studies are related to VC, especially desktop VC, which will be discussed in section 2.3.1.

Skype [1] is a very popular VoIP client developed by KaZaa in 2003. It supports voice conferencing and point-to-point video chart. Salman A. Baset and Henning Schulzrinne [2] analyzed key Skype functions such as login, call establishment, media transfer and conferencing under different situations. Their analysis was performed by careful study of Skype network traffic. The results have shown that Skype arranges voice conferencing as a centralized P2P network.

M. Reha Civanlar [3] presented an algorithm for a distributed P2P multipoint videoconferencing with a single video input/output for participants. They employed a fully distributed model via the spanning tree configuration and management of conference participants for the dissemination of media content. They also tested the feasibility of this system. The result shows that their protocol and prototype demonstrates the feasibility of such method.

K. Salah [4] proposed an analytic approach for deploying Desktop VC in a small enterprise. His approach utilized queuing network analysis and investigates delay and bandwidth for videoconferencing. But this paper only considered the point-to-point VC.

Richard Spiers and Neco Ventura [5] presented research on two different architectures for an IMS-based video conferencing system. They implemented S/C model and P2P model, and measured their signaling and data traffic overheads. The results show that the S/C model offers better network control together with a great reduction in signaling and media overheads, whereas a P2P implementation allows great flexibility, but at the expense of higher overheads.

H. Horiuchi, N.Wakamiya, and Murata [6] proposed a novel method to construct and manage a P2P network for a scalable video conferencing system. They put forward a tree reorganization mechanism and a failure recovery mechanism. From the result of the simulation experiments, they claimed that their tree reorganization mechanism can offer smooth video conferencing.

Yubing Wang, Mark Claypool, and Robert Kinicki [7] compared two different modes under various network conditions and video contents with VQM.

Ira M. Weinstein [8] described ISDN-based videoconferencing. He explained two different flavors in ISDN services, the basic rate interface (or BRI) and the primary rate interface (or PRI). He discussed the configuration and cost for different scale (small office, medium-size office, large office and enterprise) ISDN-based VC.

Most of these related works presented lots of various Multipoint Desktop videoconferencing technologies and applications. But few of them compared the different type of DVC applications and measured their QoE. It is important to be considered, since these results will help the researcher or developer deploy flexible and interactive MP DVC applications, which can provide clients good Quality of Experience (QoE).

1.4 Thesis Outline

The rest of this thesis is organized as following: Chapter 2 is the literature survey. Existing VC technologies and lots of available VC applications will be introduced and classified. In Chapter 3, we describe how to set up the groups of experiments to study their mechanisms and structures. Chapter 4 presents how we design the experiments to measure the different aspects of QoE, and we will show and analyze the results in this chapter. Chapter 5 is the conclusion of our results and Chapter 6 gives dissection the future work.

2

Survey of Multi-point Videoconferencing on the Internet

2.1 Technology of VC

According to the study of VC on the Internet, the core technologies of VC are Compression, Data delivery, Media synchronization and Conferencing architectures.

2.2.1 Media Compression

In VC, audio and video data must be transmitted in real time. Therefore, a high bandwidth is required. But only high bandwidth is not enough. For example, 'true color' needs 24 bits per pixel. A full screen image might be 640x480 pixels, over 7 million bits. For full motion video, the image is refreshed 25 times per second. This adds to over 184 Mbit/second. It is not realistically possible to transmit the information at this rate on the Internet, so video compression is required. The hardware or software performs the compression called codec. The mostly used video codec standards in VC are H.261, H.263, H.264, and MPEG-4. H.261 and H.263 codecs were developed by the ITU (International Telecommunications Union). H.264 was jointly developed by the ITU and ISO/IEC (International Organization for Standardization/International Electrotechnical Commission). MPEG-4 was developed by the ISO/IEC [9]. Now compression rates of up to 1:500 can be achieved.

2.2.2 Data Delivery

There are several ways of data delivery in VC. The mainly used now are (1) ISDN, (2) IP network and (3) satellite broadcast.

1. Integrated Services Digital Network (ISDN) [8] is offered by many telephone companies that provides fast, high-capacity digital transmission of voice, data, still images and full-motion video over the worldwide telephone network.

   ISDN is rapidly growing in popularity and is widely accepted in industry as the way to access multimedia over a network. Although it is still expensive when compared to a standard line, particularly for primary rate access, it may be suitable for inter-site conferencing.
2. Internet Protocol (IP) is the protocol used for communicating data across the Internet. Since International Telecommunications Union (ITU) approved the H.323 [10] transmission standard in 1996, and Session Initiation Protocol (SIP) was approved by IETF in 2002. Videoconferencing over IP has become more widely accepted with each passing year.
3. Satellite transmission is usually used for one-to-many conferences. Although it is expensive, cost will not be affected by distance. Therefore, it can be used where large distances or many sites are involved.

2.2.3 Media Synchronization

Media synchronization is also called audio/video synchronization, which means the image of a speaker must match the sound of the spoken words. Without the mechanism to ensure A/V synchronization, audio often plays ahead of video, because the latencies for processing video frames are bigger than the latencies of audio.

Lots of applications use Real-time Transport Control Protocol (RTCP) for media synchronization. The method of synchronizing audio and video is to delay the video and consider the audio stream as the master [11]. The receiver must receive at least one RTCP packet for each stream, to associate each RTP timebase with the common Network Time Protocol (NTP) timebase of the sender. When a video frame arrives at the receiver with an RTP time stamp RTPv, the receiver maps the RTP time stamp RTPv to the video device time stamp VTB using four steps, as illustrated in Figure 2.1.

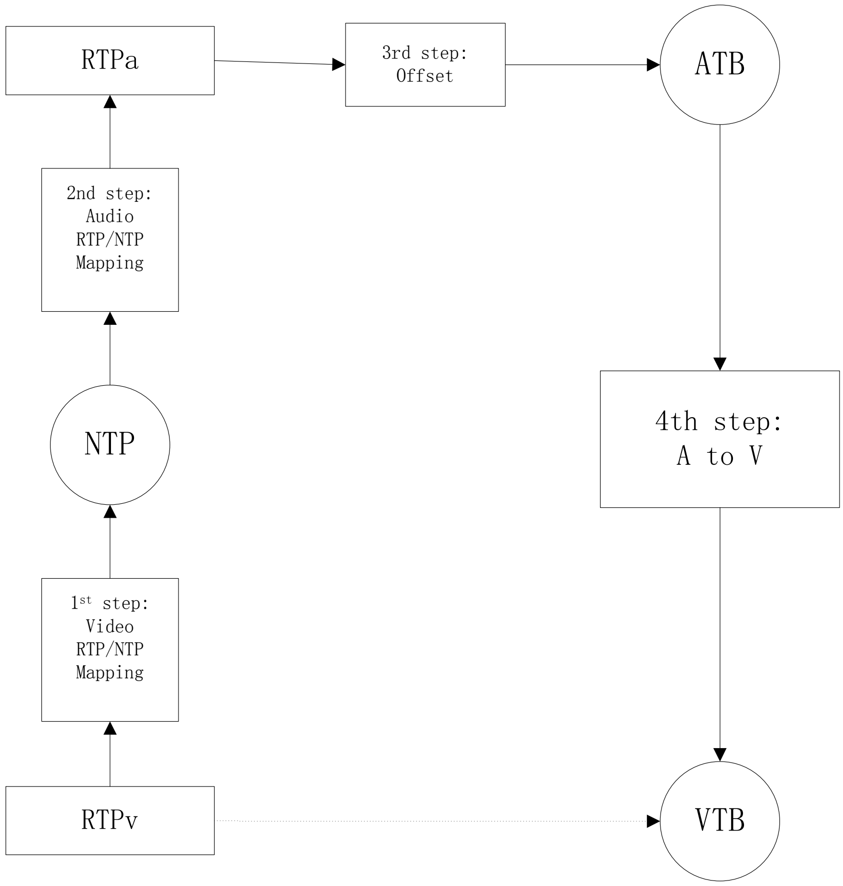


Figure 2.1 Audio and video Synchronization

After four steps, the receiver now ensures that the video frame with RTP time stamp RTPv will play on the video presentation device at the calculated local video device timebase (VTB).

2.2.4 Conferencing architectures

Conferencing architecture can be classified into two basic models: centralized and distributed.

In a centralized model (see Figure 2.2), all the components of a conferencing system are implemented in a single server (or some servers). Each endpoint only communicates with the servers.

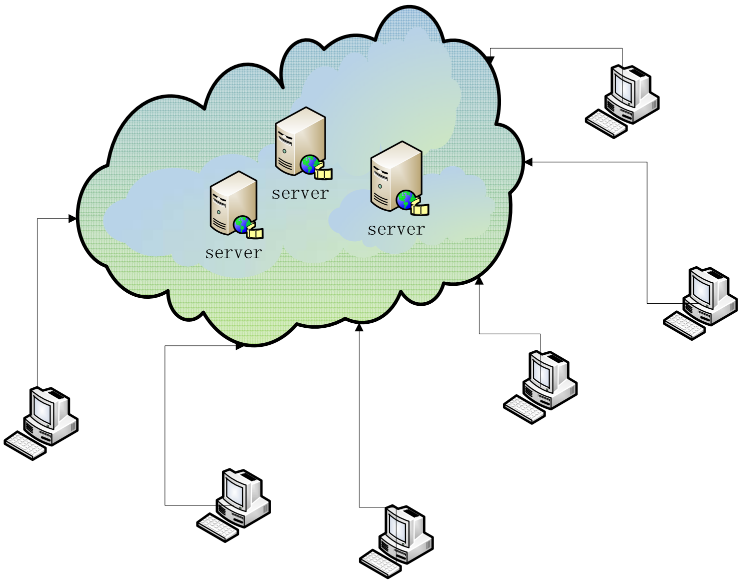


Figure 2.2 Centralized model VC

In distributed architectures (see Figure 2.3), each service provides a logical functionality distributed among multiple physical devices. There is a server or many servers for signaling, while the media data is delivered directly among endpoints.

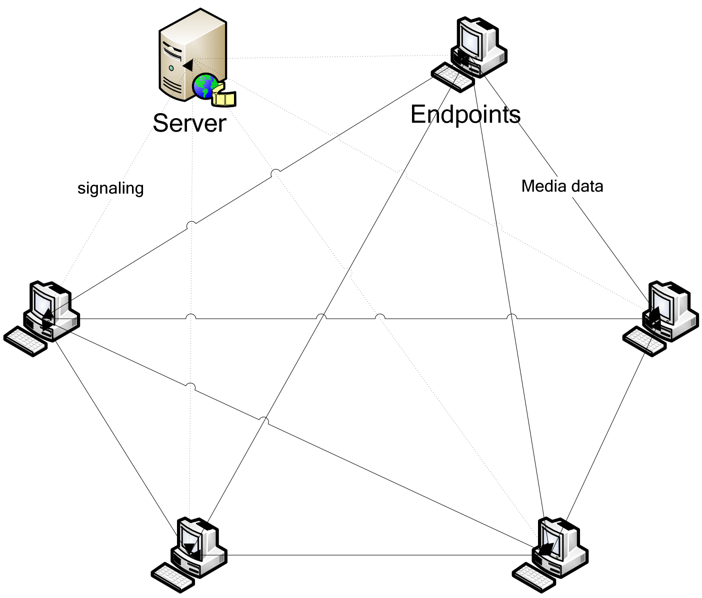


Figure 2.3 Distributed model VC

2.2 Standards of VC

The International Telecommunications Union (ITU) has three main umbrellas of standards for VC [13].

1. ITU H.320 is known as the standard for public switched telephone networks (PSTN) or VTC over integrated services digital networks (ISDN) basic rate interface (BRI) or primary rate interface (PRI). H.320 is used on dedicated networks such as T1 and satellite-based networks. Business, government and military organizations still predominantly use H.320.
2. ITU H.323 is a standard for videoconferencing over non-guaranteed quality-of-service packet network, such as LANs, Internet. H.323 has the advantage that it is accessible to anyone with a high speed Internet connection, such as DSL.
3. ITU H.324 is the standard for transmission over POTS, or audio telephony networks. It is a compression standard that enables video conferencing systems to achieve highly error resilient IP video transmission over the public Internet without quality of service enhanced lines. This standard has enabled wide scale deployment of high definition desktop video conferencing and made possible new architectures which reduce latency between transmitting source and receiver, resulting in fluid communication without pauses.

2.3 Classification of VC

The VC can be classified from different point of view.

2.3.1 According to the different terminals:

a) Desktop VC: Add application to a computer with Internet. Convenience and low-cost, but quality cannot be guaranteed. In this thesis, we will focus on this kind of VC systems.
b) Room VC System: Special room with professional equipments. It has high quality video and audio, but it is fixed and very expensive. In TU Delft, there is a Room VC system located in Mekelweg 2 Blok 8A. It uses POLYCOM HDX9000 videoconference system, which costs 40000 USD, and 125 euro/hour to use. It can support six participants simultaneously.
c) Videophone: Only point-to-point communication.

2.3.2 According to the user interface

User interface is the connection between the user and the VC systems. It enables the user to log in, schedule new meetings, attend meetings, and get some in-conference controls. In general, there are two kinds of user interface:

a) Web browser interface: Users can schedule new meeting, join existing meeting through a web browser without installing any software.

b) Software user interface: The interface generally consists of a series of menus, allowing the caller to interact with the system based on a set of context-sensitive scripts.

2.3.3 According to the signaling protocol used:

a) Based on H.323

H.323 is an umbrella Recommendation from the ITU Telecommunication Standardization Sector (ITU-T) that defines the protocols to provide audio-visual communication sessions on any packet network. The H.323 standard addresses call signaling and control, multimedia transport and control, and bandwidth control for point-to-point and multi-point conferences.

H.323 system (see Figure 2.4) defines several network elements including endpoint, gateways to allow interworking between IP network and other network types.

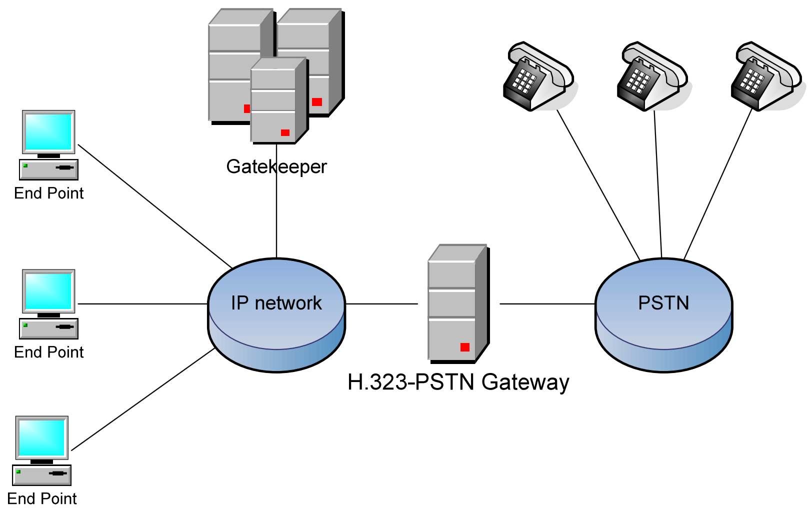


Figure 2.4 H.323 Networks

b) Based on SIP

The Session Initiation Protocol (SIP) [22] is a TCP/IP-based Application Layer protocol. It was designed by Henning Schulzrinne and Mark Handley in 1996. The latest version of the specification is RFC 3261from the IETF Network Working Group. The SIP network is shown in Figure2.5.

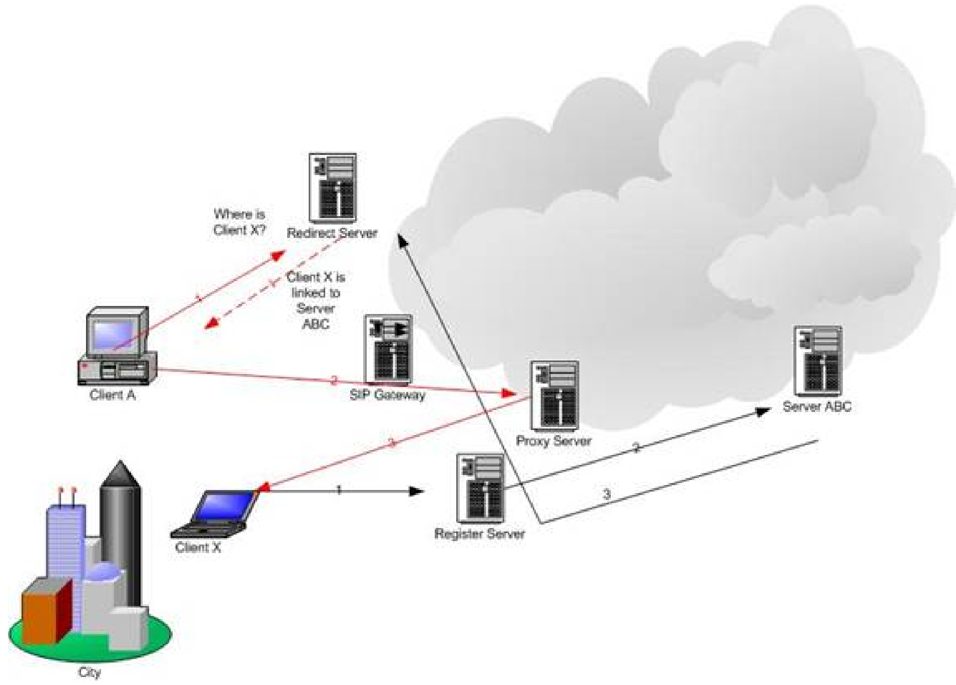


Figure 2.5 SIP Networks

2.3.4 According to the network structure:

a) Centralized network

It is also called Client/Server network.

b) Distributed network

It can be further divided into:

1. Decentralized P2P network.

2. Hybrid (Centralized and decentralized) network.

2.4 Available VC applications

In order to choose suitable VC applications for further detailed study, we made a survey of the available Desktop VC on the Internet.

2.4.1 Centralized (Client/Server) network

- Mebeam is a C/S structure MP videoconference service. It support 16 people in a multiuser-videochat at most and not required to use the same IM network.
- Confest provides the C/S MP videoconference service, which is written in C and C++ in client and server. Confest uses point-to-point asynchronous transmission. The server transfers data directly to each client. Each video and audio data packets reached the maximum TCP/IP data packet, which is 1640 bytes. Confest supports

15 video people and unlimited audio people at the same time.

- Himeeting uses C/S structure and supports digital whiteboard, presentations, application sharing, text chat, voting, web video, and provides automatic bandwidth management to optimize the use of all types and qualities of network connection.
- EarthLink provides C/S Multipoint VC service, which has two communication ways: conference sessions and private chat. It supports Multi-protocol such as AIM, Jabber, and IRC.
- IOMeeting is Brower/Client Multipoint videoconference software, which uses web2.0 technology AJAX.
- VideoLive provides Web-based Multipoint videoconference service and up to 6 people can be seen at one time during a VC.
- VidSoft is Client/server video conferencing software. It allows audio, video and data communication over the Internet or corporate intranet and allows up to ten endpoints to meet simultaneously.
- MegaMeeting Video Conferencing Offers a range of VC services, from personal to enterprise level. Guest participants need only a web browser with Flash. There are no additional software to install. It supports up to 16 live video streams simultaneously.
- Nefsis uses a massive network of distributed computers to ensure the VC service. There are many servers (referred as access point) all over the world. Once connected to an access point, Nefsis leverages the built-in load balancing and failover abilities of its distributed cloud to improve overall video conferencing performance. It supports up to 10 live video streams simultaneously.
- CloudMeeting is a Real-Time multimedia Internet group communications service. CloudMeeting supports up to 6 video feeds at a time. If bandwidth is low or fluctuating, CloudMeeting dynamically adjusts the number of frames per second at the end user in order to keep the meeting and audio running smoothly.
- WebEx Meeting Center is VC software developed by Cisco. It supports unlimited meetings with 25 people and up to 6 video streams at a time.

2.4.2 P2P videoconference

- Vsee is decentralized P2P structure MP videoconference software. It supports up to

4 people at the same time.

- Qnext is centralized P2P network multi-point videoconference software. The node which hosts the meeting is the center node. Qnext supports 8 people at most at one time and 4 people with video stream.
- Vmukti BizCom is Multipoint video and audio media call center software. It uses P2P private branch exchange (PBX) and the number of the users depends on the bandwidth. The architecture of the Vmukti BizCom is the tree. Father node is the supper node in a conference.
- Lava-Lava is a P2P VC application, which provides Multipoint VC service and can support 5 people at the same time to communicate with video and audio.

The survey about their max frame rate, maximal users and the structure is summarized in the Table2.1.

	Max. frame rate(frames/second)	Max. number of video feeds	C/S or P2P
Eedo WebClass		6	C/S
Linktivity WebDemo	30	6	C/S
IOMeeting	30	10	C/S
EarthLink	30	24	C/S
VideoLive	30	6	C/S
Himeeting	17	20	C/S
VidSoft	30	8	C/S
MegaMeeting	30	16	C/S
Nefsis	30	10	C/S
Smartmeeting	15	4	C/S
Webconference.com	15	10	C/S
Mebeam		16	C/S
Confest	30	15	C/S
CloudMeeting	30	6	C/S
WebEx	30	6	C/S
Lava-Lava	15	5	P2P
Vmukti	30		P2P
Qnext	30	4	P2P
Vsee	30	8	P2P

Table2.1 Survey of the Available VC applications on the Internet

3

Lab Experiments

In this chapter, we analyzed four typical Internet multipoint videoconferencing systems: Mebeam, Nefsis, Qnext, and Vsee. These systems have worldwide downloads and use. When studying these applications, lots of challenges came out. All of these systems are used commercially, which means their source code is not available. In order to get more insight of the mechanism and behavior of these systems, we designed some experiments to capture the data, and tried to analyze the data in different scenarios. Because all the applications have a Graphical User Interface (GUI) and have to establish the meeting manually, we have to access the nodes physically, not in Linux platform or with remote control (like Planetlab). An overview of how these multipoint VC systems work will be presented below.

3.1 Experiment Environment

For our lab experiments, there are four computers with Window XP used in the lab of the Telecommunication Department. “E2eSoft VCam” is used as virtual camera, and “Wireshark” is used for capturing the data at each participant. Each computer uses a 10/100 FastEthernet network interface, which connected to a 100Mbit switch in the lab, which further connect to a router of SurfNet. The measurement infrastructure is shown in Figure 3.1.

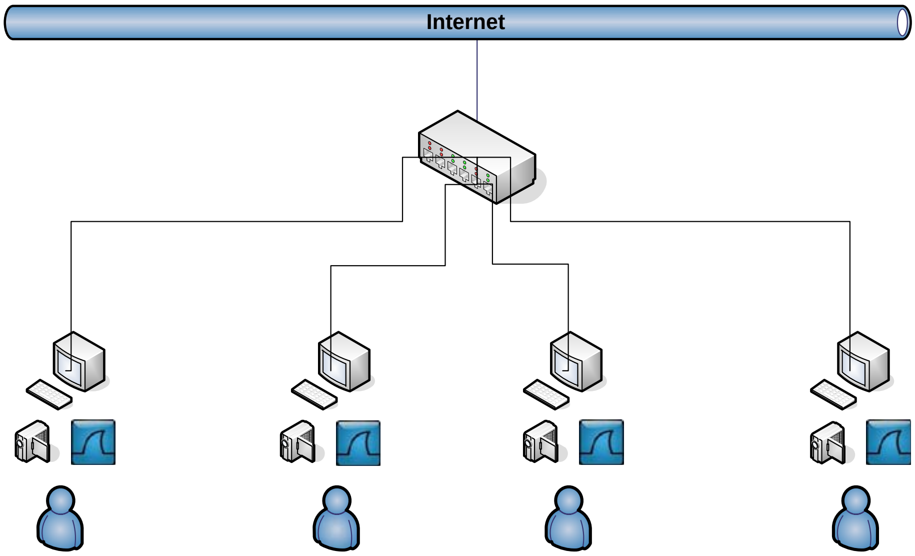


Figure 3.1 Measurement infrastructure

3.2 Measurement/Methodology

3.2.1 Measurement applications

In order to measure the Desktop VC on the internet, we chose four applications to measure.

1. Mebeam

Mebeam [14] (Figure 3.2) is a web-browser based multipoint VC application, which can support 16 people in a VC at most.

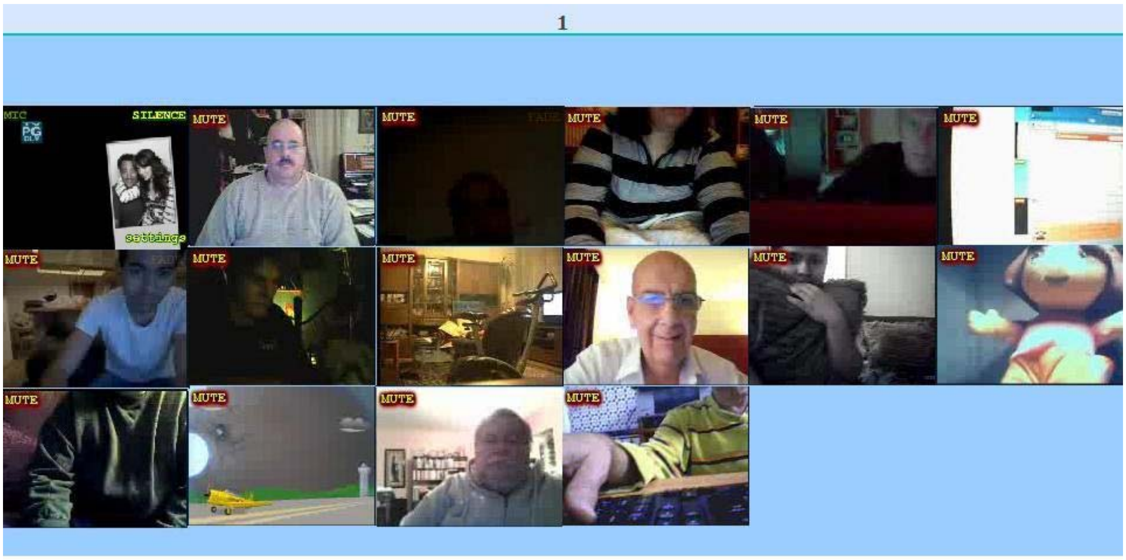


Figure 3.2 Mebeam user interface

2. Nefsis (Free trial version)

Nefsis [15] (Figure 3.3) is a video conferencing system designed by Nefsis Corp. It uses a massive network of distributed servers (claimed as cloud) to ensure the quality VC conferencing experience.

Figure3.3 Nefsis user interface

3. Qnext (version 4.0.0.46)

Qnext [16] (Figure 3.4) is P2P network multipoint VC software developed by Qnext Corp. It has windows and Linux versions. Qnext supports 4 people for VC at most simultaneously.

Figure 3.4 Qnext user interface

4. Vsee (version 9.0.0.612)

Vsee [17] (Figure 3.5) is also P2P structure MP VC software, which supports up to 8 people at the same time. It runs on Windows 2000, XP, Vista, and Windows 7. Communication among users is first authenticated and controlled via a directory server, and then travels peer-to-peer over a single UDP port or via HTTP or SSL tunneling if a direct connection is not possible.

Figure 3.5 Vsee user interface

3.2.2 Methodology

There are two ways of data collection and analysis techniques: active measurement and passive measurement.

Active measurements require injecting test packets into the network, such as Ping and Traceroute using ICMP packets. Compared with active measurement, the passive measurements do not inject test packets into the network. They capture packets from network devices (routers or switches).

In this chapter, we use passive measurements to collect data for analyzing the multipoint videoconferencing mechanism. In the part of Quality of Experience experiments, we use both active and passive measurements to get the results.

3.3 Results

In order to understand the mechanism of these applications, we divided the experiments into five parts: (1) login; (2) call establishment; (3) I/O traffic; (4) packet distribution and (5) the topology. The details of the result are presented below.

3.3.1 Login and Call establishment

3.3.1.1 Mebeam

We used Mebeam to build a web video-chart room and other participants entered in. We captured the packets at each participant by wireshark. After analyzed the source and destination of the packets, we can conclude the login and call establishment processes of Mebeam clients (can be seen in Figure 3.6).

Client: A, B, C; Login server: 66.63.191.202; Conferencing server: 66.63.191.211

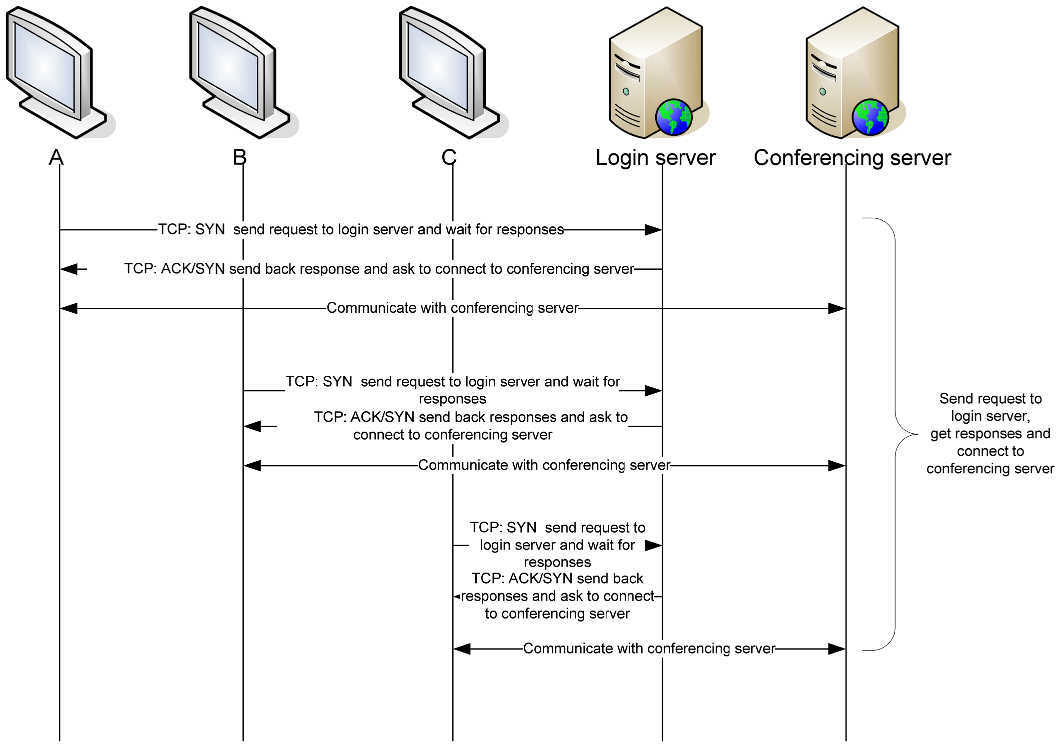


Figure 3.6: Mebeam login and communication architecture

In Mebeam, before the conference establishes, each client sends requests to the login server first. After getting the responses, all clients connect to the conferencing server. Each client only communicates media data with the conferencing server, there is no connection among the participants.

3.3.1.2 Qnext

When participants log in the Qnext Client and set up a conference, we use wireshark to capture the packets at each participant. According to the packets data flow, the login processes of Qnext clients can be seen in Figure 3.7.

Login part: Client: A, B, C; Login server: 207.99.120.136, 207.99.120.135

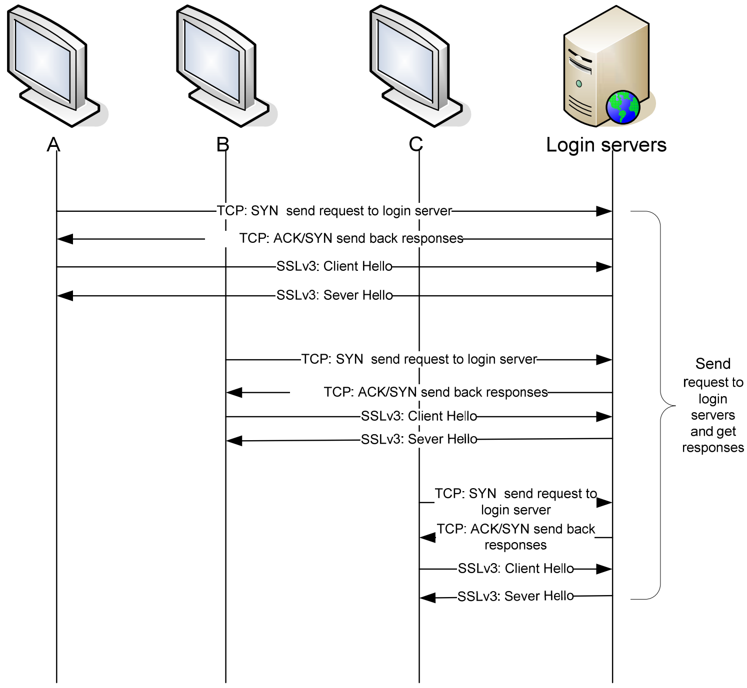


Figure 3.7: Qnext login architecture

In Qnext login part, each client sends TCP packets to the login server. After getting response, the server and client use SSLv3 protocol to send messages (client hello and server hello) to connect the login server.

The call establishment processes of Qnext clients can be seen in Figure 3.8.

Signaling servers: 216.235.57.72, 75.105.198.36, 79.118.145.228

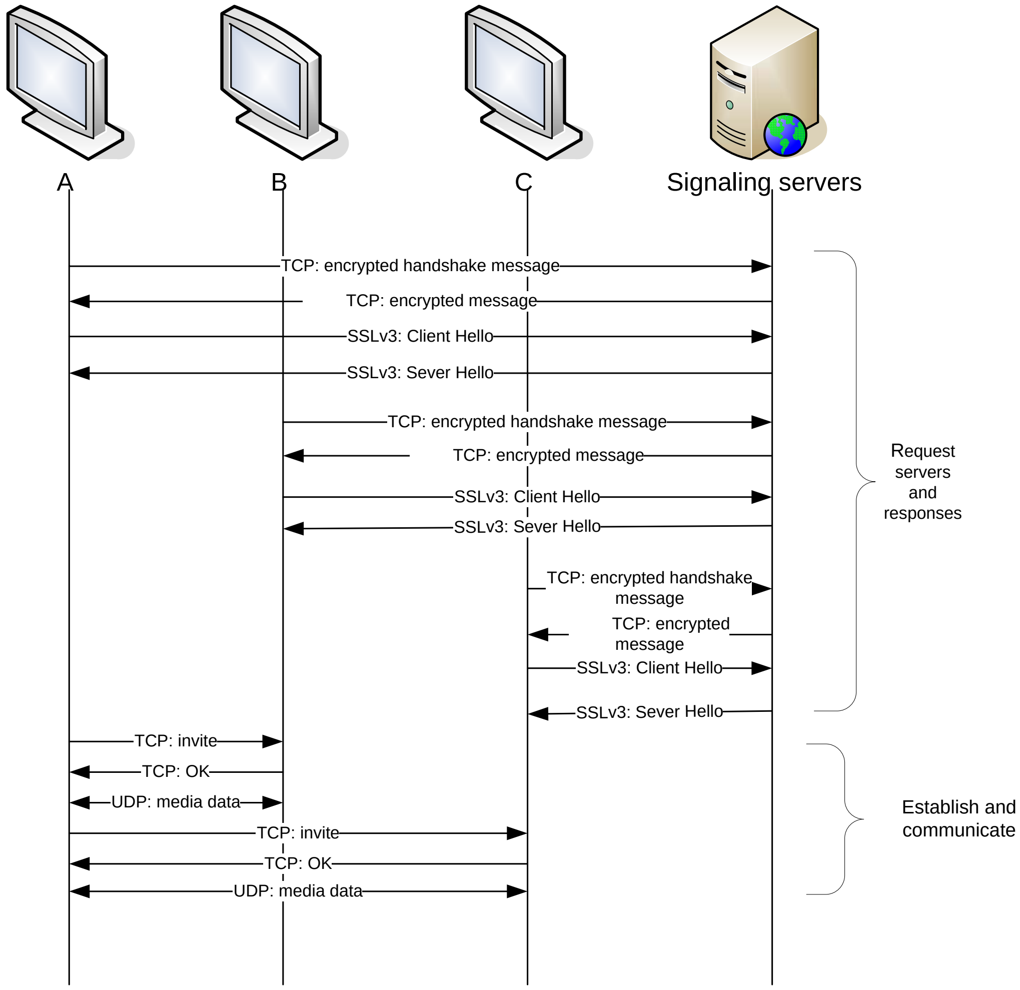


Figure 3.8: Qnext conference communication architecture

In Qnext, before call is established, each client will exchange encrypted handshake messages with signaling servers by TCP packets first, and then use SSLv3 protocol to send connection information messages (client hello and server hello) between the client and the signaling server. If one client accepts the other client's request, it uses User Datagram Protocol (UDP) to transfer media data between clients. In a conference, there is only one host, and only the host can invite other clients. Other clients can only communicate with the host.

3.3.1.3 Vsee

According to the packets we captured by wireshark, the Login and call establishment processes of Vsee clients are described in Figure 3.9.

Client: A, B, C

Web servers: 216.139.215.232, 85.12.57.89, 208.83.212.12, 208.69.180.146, 67.210.110.36, 66.235.214.72, 66.235.209.110, 67.15.62.14, 216.139.215.232

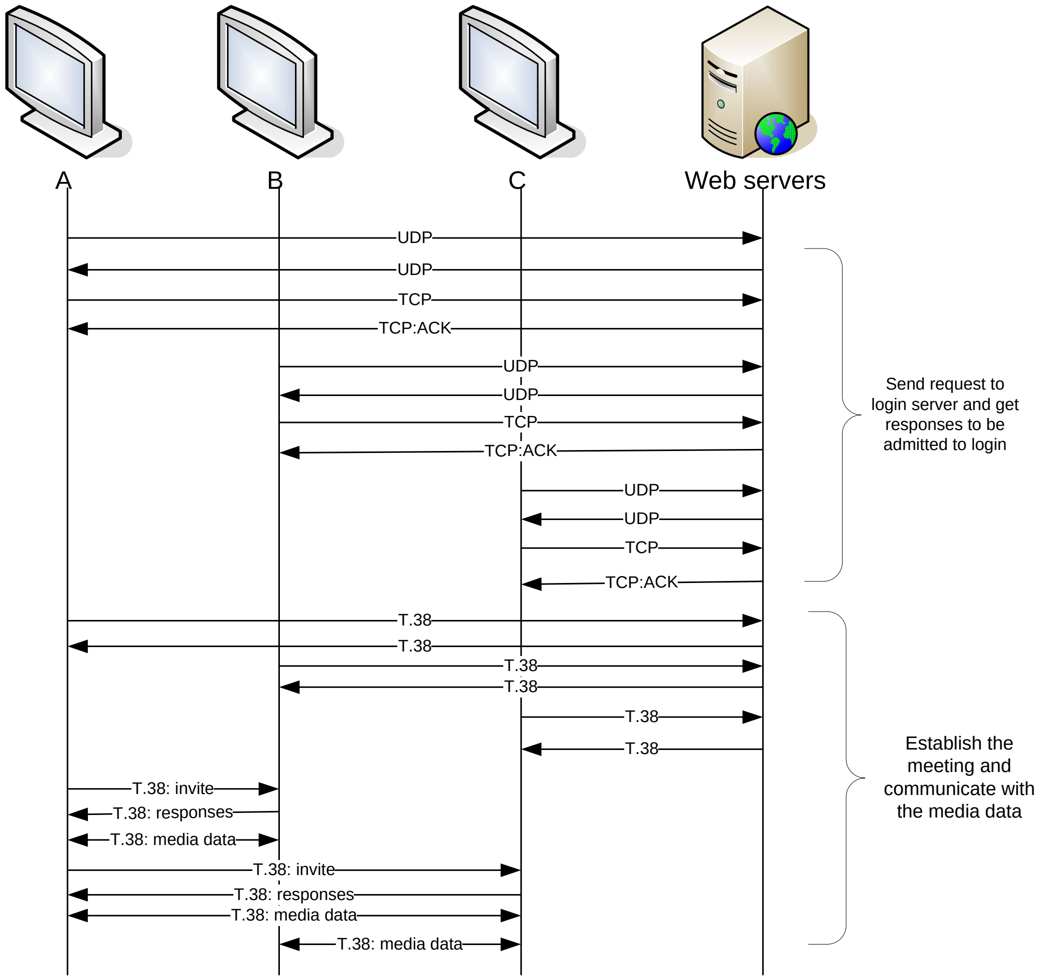


Figure 3.9: Vsee login and conference communication architecture

In Vsee, participants sent UDP and TCP packets to communicate with web servers and identified by the servers in login process. In call establishment process, T.38 was used to sending and receiving the conference host invitation. After responses are received, the media data will be transferred among participants. T.38 is a protocol of ITU for allowing transmission of fax over IP networks in real time. During the conference, all the traffic is encrypted with FIPS (Federal Information Processing Standards) 140-2 256 bit AES (Advanced Encryption Standard).

3.3.1.4 Nefsis

When we use Nefsis to set up a conference, wireshark was used to capture the packets. Based on the packets data flow, the Login and call establishment processes of Nefsis clients are shown in Figure 3.10.

Client: A, B, C;

Nefsis virtual conferencing servers: 128.121.149.212, 118.100.76.89

Access points: NL, Rotterdam node, 213.163.84.51; UK, Blueconnex, 92.48.74.10;

India, 203.199.75.35; India, 116.240.200.117; Singapore, 210.193.54.215; NL, Amsterdam, 92.48.92.21

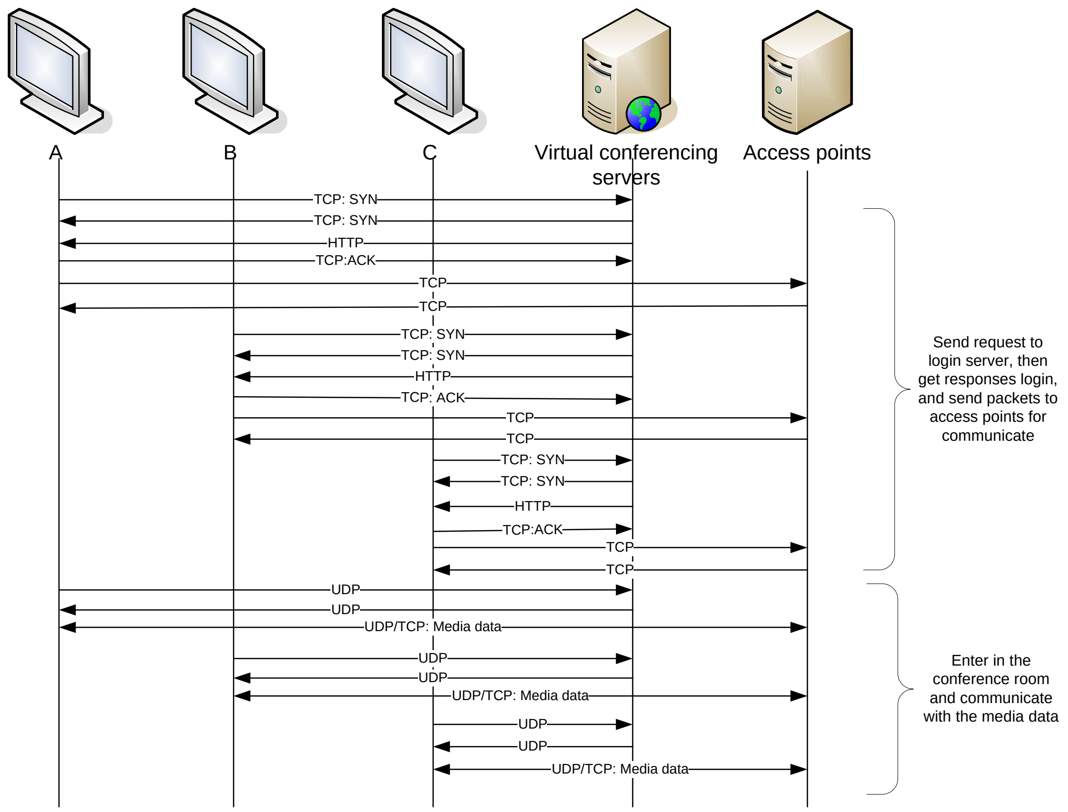


Figure 3.10: Nefsis login and communication architecture

In Nefsis, they have lots of access points (conferencing server) all over the world. Each client uses TCP and HTTP to login to the virtual conferencing servers and get responses for identification. The access points will choose some access points for the participant by sending TCP packets. If the participant wants to join a conference, a request must be sent to the virtual conferencing server. After get the responses, the participant could enter in the conference room and communicate with media data. If there is a firewall, each client communicate with each other through the access point with TCP, otherwise these clients can communicate directly with UDP. The meeting host can select the option of SSL/TLS Security, which uses encrypted end-to-end connections.

All of these Desktop VC applications are not open source. They are black boxes for us. However, according to data we gathered and analyzed above, we knew how these applications login to the systems and how they established the meeting through the signaling and media packets. It is helpful for us to learn their behaviors and mechanisms, and also important for investigating their structures to set up connections and deliver the stream content.

3.3.2 I/O traffic

In order to understand the mechanism and traffic load of the videoconferencing application, it is important to analyze the I/O traffic. We set up meetings for different applications and all the applications use the same video stream.

Table 3.1 shows the upload and download I/O traffic for four participants in different applications.

Applications / I/O traffic		Qnext	Vsee	Nefsis	Mebeam
Jackstraw (Host Qnext)	Total	1.243Mb/s	0.843Mb/s	0.640Mb/s	0.129Mb/s
	Upload	0.858Mb/s	0.486Mb/s	0.162Mb/s	0.095Mb/s
	Download	0.385Mb/s	0.357Mb/s	0.478Mb/s	0.034Mb/s
Lzh	Total	0.421Mb/s	1.098Mb/s	0.629Mb/s	0.102Mb/s
	Upload	0.111Mb/s	0.453Mb/s	0.205Mb/s	0.060Mb/s
	Download	0.310Mb/s	0.545Mb/s	0.424Mb/s	0.042Mb/s
Sfc	Total	0.397Mb/s	0.824Mb/s	0.542Mb/s	0.132Mb/s
	Upload	0.108Mb/s	0.439Mb/s	0.157Mb/s	0.045Mb/s
	Download	0.289Mb/s	0.385Mb/s	0.385Mb/s	0.088Mb/s
Yuelu	Total	0.425Mb/s	0.794Mb/s	0.553Mb/s	0.166Mb/s
	Upload	0.129Mb/s	0.439Mb/s	0.163Mb/s	0.106Mb/s
	Download	0.296Mb/s	0.355Mb/s	0.390Mb/s	0.056Mb/s

Table 3.1 upload and download I/O traffic for different applications

From this table, we can see the traffic load of Mebeam is much less than four other applications. Besides that, the traffic load of Qnext host is much more than other participants. The reason is the host of Qnext act as a super node, and other participants can only communicated with the host. Figure 3.11 shows the I/O traffic for each application from two users to four users:

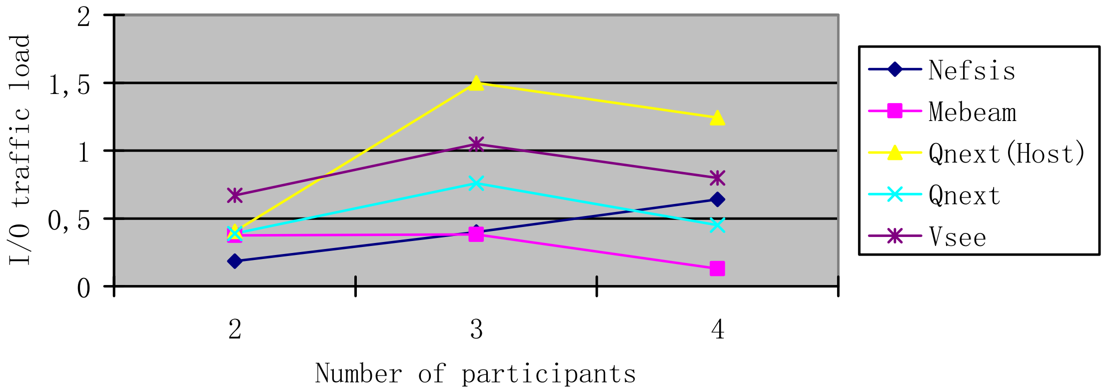


Figure 3.11 I/O traffic for each application from two users to four users

We can know from Figure 3.11 and Table 3.1 that Qnext has the heaviest I/O traffic, especially the host client. Mebeam has the lightest I/O traffic. In Figure 3.11, the decreasing slope after three users indicates that when the number of participants increased, all VC applications except Nefsis re-coded the stream in order to control the I/O traffic. We also did an experiment for Nefsis in five participants. The I/O traffic load is 0.586Mbit/second, which means after five users Nefsis also re-coded the stream.

3.3.3 Packet size distribution

In order to distinct the signaling, video and audio packets, we did three experiments for each application. Firstly, we used two computers with cameras and microphones. Then, we used two computers only with microphones but no camera. And last, we used two computers neither with microphone nor camera.

Packet size distribution of Mebeam is given below.

1. With both microphone and camera:

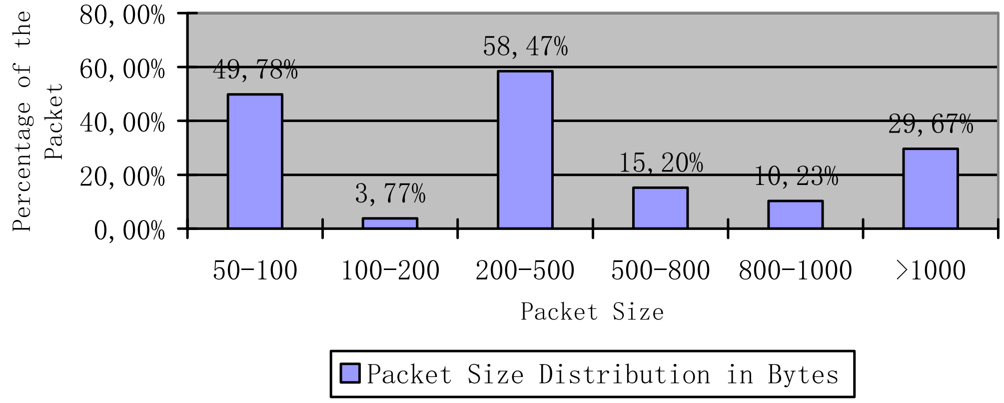


Figure 3.12 Mebeam media packet distribution in experiment 1

2. Only with microphones but no camera:

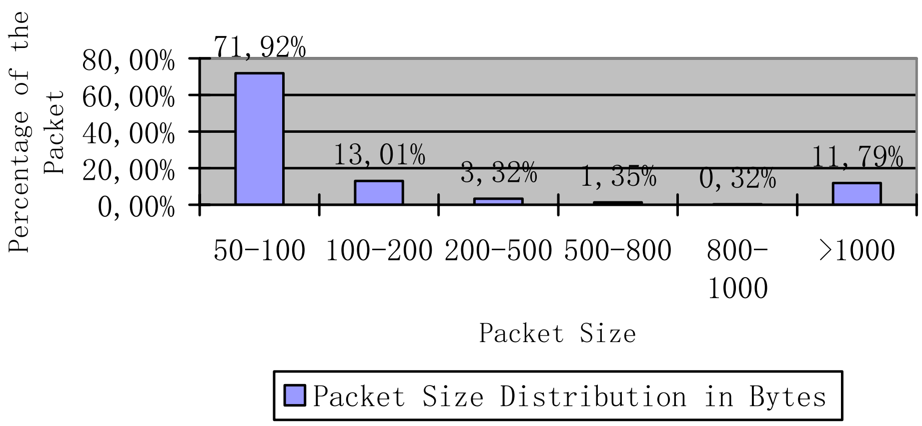


Figure 3.13 Mebeam media packet distribution in experiment 2

3. Neither with microphone nor camera

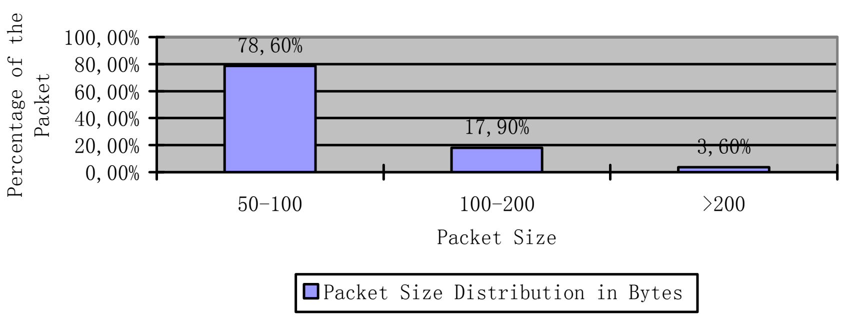


Figure 3.14 Mebeam media packet distribution 3

From the results above, we can see that the size of video packets and audio packets are not fixed. The video packets are above 200 bytes.

Similarly, we can get the packets size distribution of Qnext, Vsee and Nefsis respectively:

- In Qnext, the size of video packets and signaling packets are not fixed, the size of audio packets is between 100 and 200 bytes, while the video packet size ranges from 50 to 1100 bytes.
- In Vsee, the size of video packet of Vsee is between 500 and 600 bytes, the audio packet size is between 100 and 200 bytes, and the signaling packet size is between 50 and 100 bytes.
- In Nefsis, the signaling packet size of Nefsis is between 50-100 bytes, the audio packet size is between 100-200 bytes, the video packets size is also not fixed, but mainly between 1000-1600 bytes.

The packets size distribution can help us plot the topology of different applications.

3.3.4 Topology

The topology describes the data transmission between participants in the VC applications.

Based on the packets data captured from the wireshark at each participant, we can get the destination and source of the packets. According to their data flow and the packet size distribution, we can plot the data distribution for these applications.

3.3.4.1 Mebeam

The topology of Mebeam is shown in Figure 3.15.

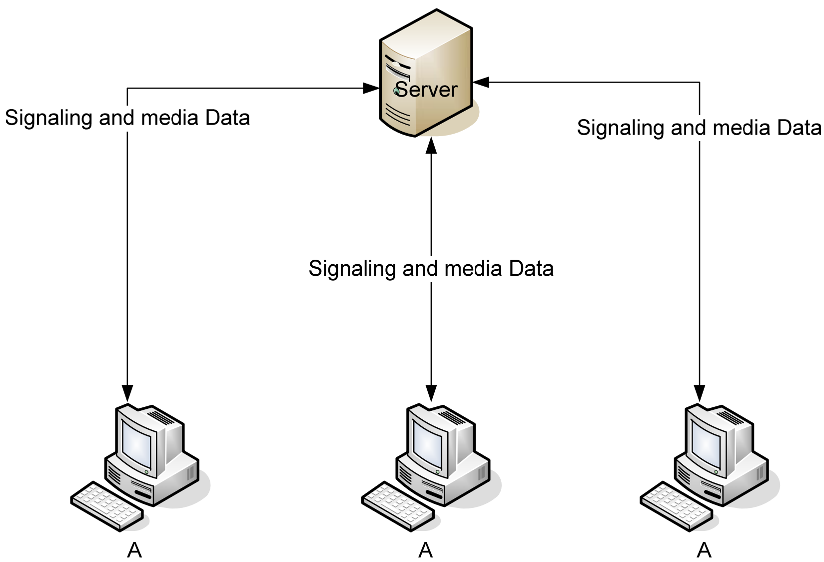


Figure 3.15 Mebeam conference topology

Each client communicates with the servers by sending signaling and media data. There is no connection among the clients.

3.3.4.2 Qnext

The topology of Qnext is shown in Figure 3.16.

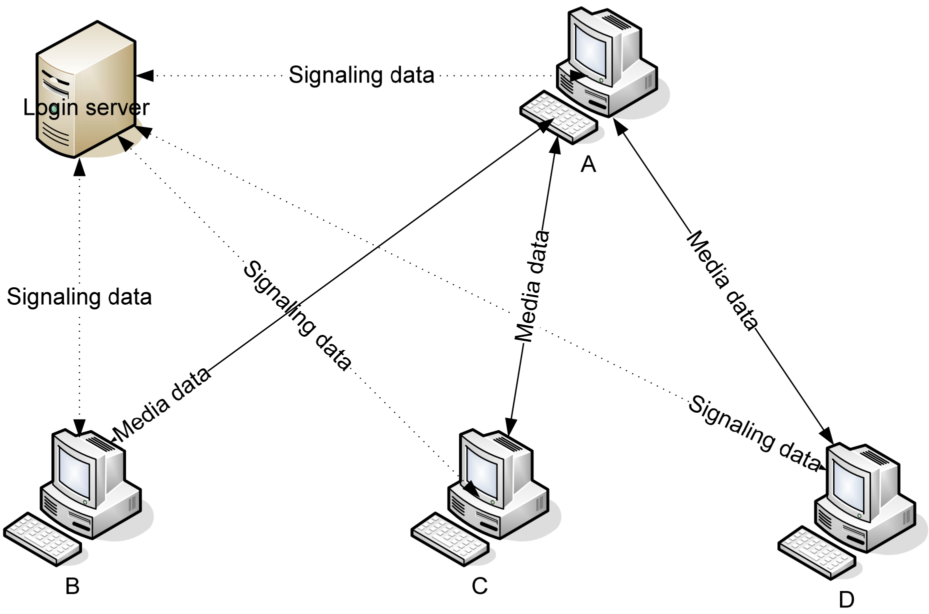


Figure 3.16 Qnext conference topology

During the conference, there is a server for processing signaling data with each client. After the call is established, there is a super node, which is the conference host. Other clients only communicate media data with the host.

3.3.4.3 Vsee

From the I/O traffic and the packets data flow, we can conclude the topology of Vsee as shown in Figure 3.17.

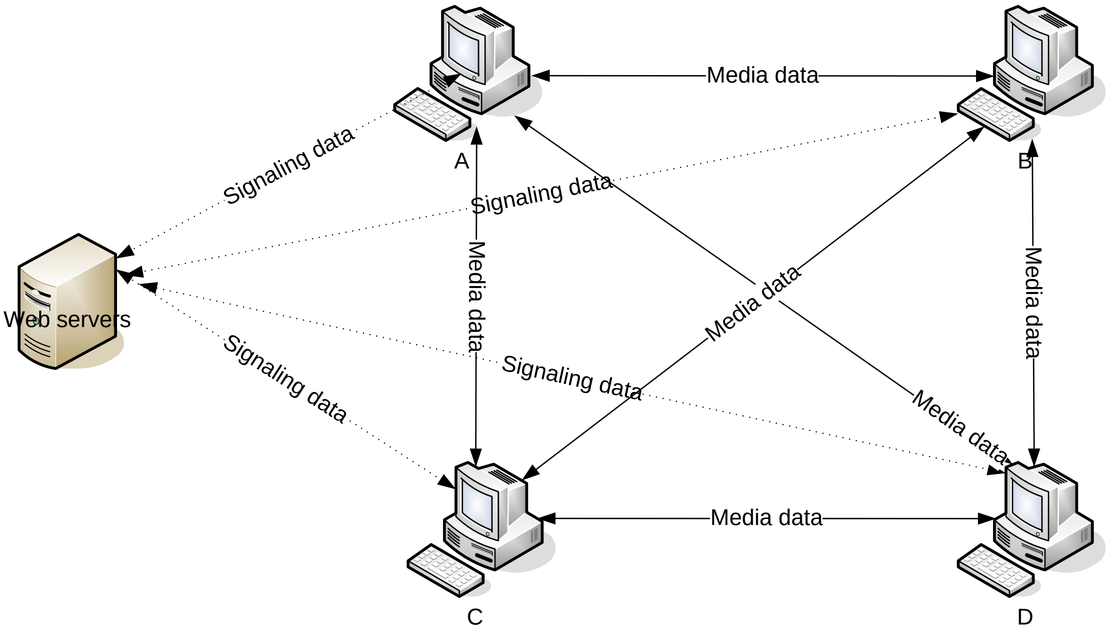


Figure 3.17 Vsee conference topology

During the conference, there are some web servers for signaling data in Vsee. After the meeting is set up, Vsee has a full-mesh topology for media data delivery. Each node has equal weight to communicate with each other.

3.3.4.4 Nefsis

Nefsis claims it has cloud architecture. In order to investigate its network structure , we made a set of experiments:

1. Two nodes: A (public IP) is in campus of TU Delft; B (public IP) is in Yue's office is also in campus of TU Delft.

   After setting up the meeting, we captured the packets from wireshark.

   A connected with B directly as shown in Figure 3.18:

Figure 3.18 Nefsis conference topology 1

2. 2-A: Two nodes: A (private IP) is in the Roland holstlaan in Delft, B (public IP) is in Yue's office in TU Delft.

From wireshark, we found A and B connected with the same node C (213.163.84.53), as shown in Figure 3.19.

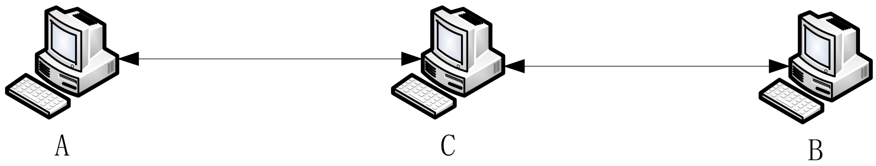


Figure 3.19 Nefsis conference topology 2

2-B: Two nodes: A (public IP) is in the lab of 19th floor in EWI, B (private IP) is in Yue's home in Rijswijk, as shown in Figure 3.20.

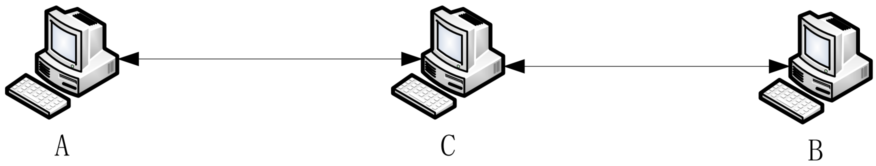


Figure 3.20 Nefsis conference topology 3

3. Two nodes: A (private IP) is in the Roland holstlaan in Delft, B (private IP) is in Wuhan, China.

   From wireshark, we found that A and B connected with different nodes E (92.48.92.21), D (125.220.137.132), as shown if Figure 3.21.

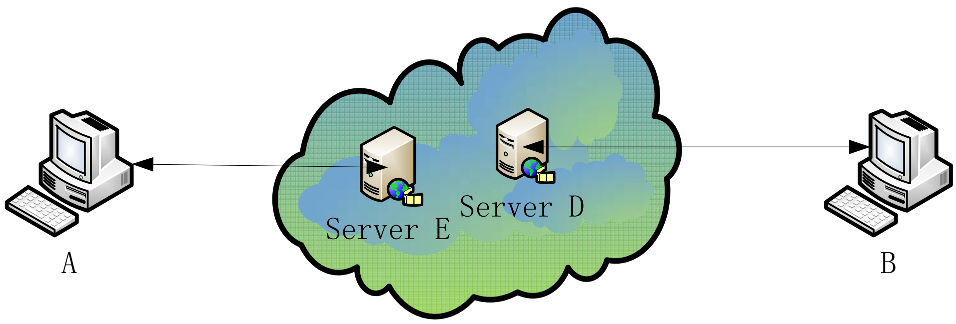


Figure 3.21 Nefsis conference topology 4

4. Three nodes: A (private IP) is in the Roland holstlaan in Delft, B (private IP) is in Wuhan, China, C (private IP) is in Changsha, China.

   From wireshark, we found that A, B and C are connected with different nodes E (92.48.92.21), D (125.220.137.132), and F (210.193.54.217), as shown in Figure 3.22.

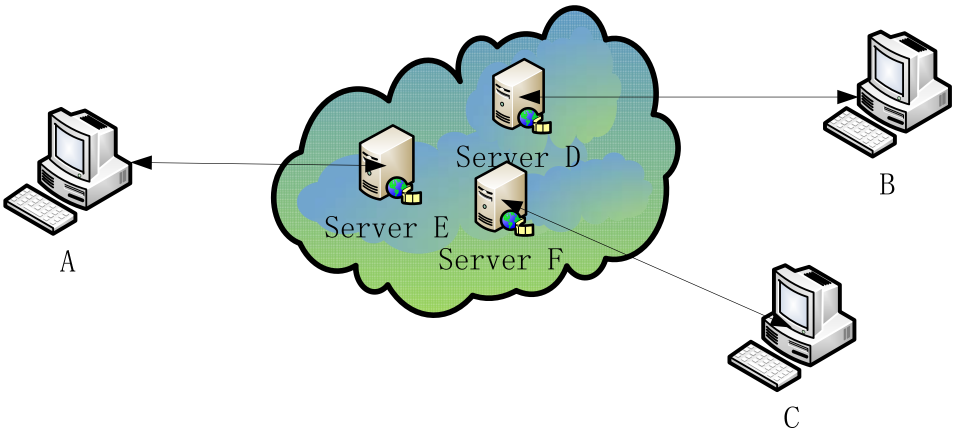


Figure 3.22 Nefsis conference topology 5

5. Four nodes: A (private IP) is in the Roland holstlaan in Delft, B (private IP) is in Beijing, China, C (private IP) is in Changsha, China, D (public IP) is in TU Delft.
   From wireshark, we found A, B, C and D all connected with the node E (210.193.54.215), as shown in Figure 3.23.

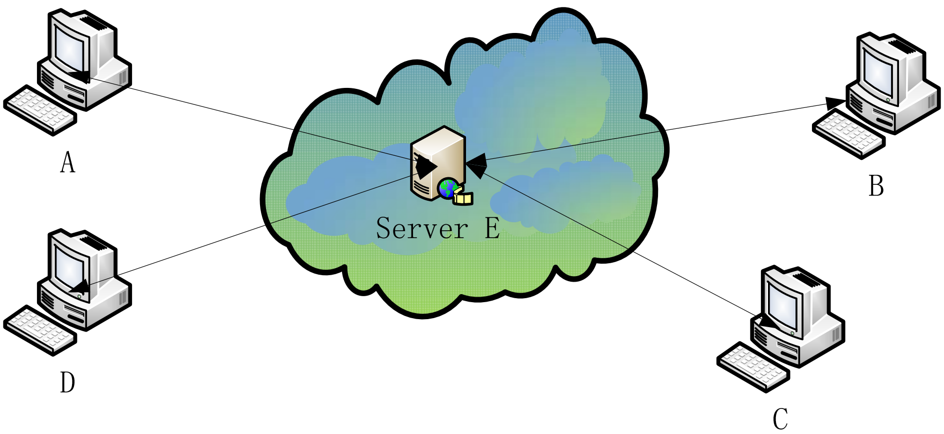


Figure 3.23 Nefsis conference topology 6

From all the experiments we did, we can conclude that when there are only two participants in the meeting and both of them are public IP, they will connect directly. If the participants are private IP, one or more conference servers will be used to transfer the media data.

We plotted the Nefsis network structure as shown in Figure3.24.

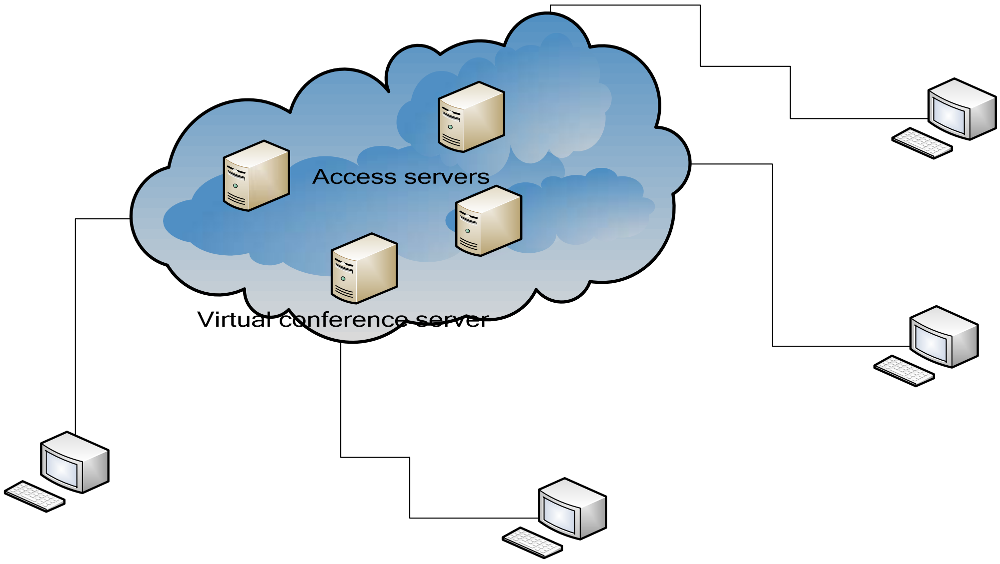


Figure 3.24 Nefsis conference topology

In the Nefsis system, there are some virtual conference servers and access servers located all over the world. If there are private IP, when the meeting was established, each client will communicate through these access servers.

Based on these experiments, we got to know the different topologies in desktop VC systems. In the next chapter, we will see how these topologies affect the Quality of Experience.

3.4 Discussion

Through these experiments, we know more about the mechanism and behaviors of different multipoint videoconferencing applications. In general, according to the result of the experiments, we can observe that:

1. In login part, in order to keep the user's information security, all measured applications except Mebeam use encrypted packets to communicate with the login server.
2. Mebeam uses TCP protocol to communicate and transfer data. Qnext, Vsee and Nefsis use UDP to transfer data during the conference.

3. We did experiment with physical camera for all applications, we found if the image in the camera changed acutely, the traffic load increased remarkably as shown in Figure 3.25 (we use Qnext for example).

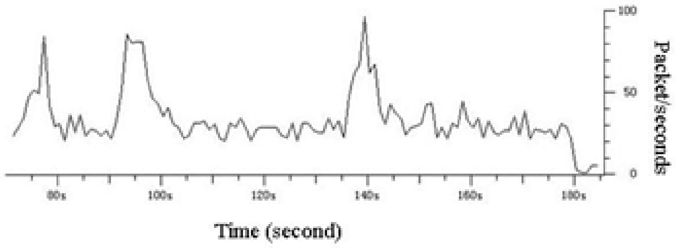


Figure 3.25 I/O traffic in a physical camera

4. When the number of participants increases, the I/O traffic will increase. But when the number of participants is up to four, Vsee, Qnext and Mebeam will re-code the stream and control the traffic load. When the number continues to increase, Nefsis will also re-code the stream.
5. For different applications, the video and audio packet sizes are different. Some applications have the fixed size of video and audio packet (i.e. Vsee). Some applications don’t have the fixed video and audio packets (i.e. Mebeam), and some applications have fixed audio but no fixed video packets (i.e. Qnext, Nefsis).
6. Different applications with different topologies generate different I/O traffic. From the general subjective opinions, they also provide different video quality, which will be analyzed in the next Chapter.
7. For some applications, the host of the meeting plays the most important role for the conference, such as Qnext and Vsee. If the host leaves, the conference cannot continue. However, for some other applications, even the host leaves, as long as there still are some participants, the conference can keep alive (i.e. Mebeam and Nefsis).

4

Quality of Experience

Quality of Experience (QoE) is used to describe the overall service quality from user's perspective. It can tell how customers feel about the service. Considering VC applications are developed with proprietary technologies, and the communication bandwidth is not always infinite and stable, testing QoE is one of the most challenging tasks that network researchers have to face in multipoint video conference systems. In this chapter, we mainly consider four factors. The first factor is speech quality. We use PESQ to measure the audio quality. The second factor is video quality. We measured the difference of video quality between any pair of sender and receiver. The third factor is the interactivity level among participants. We designed some experiments to measure the conversation delay. We also considered the synchronization between audio and video tracks of the media as the forth factor.

4.1 PESQ

PESQ (Perceptual Evaluation of Speech Quality) is a standard for objective voice quality testing that predicts the results of subjective listening tests. It is developed by KPN and British Telecommunication in 2000 and it is standardized as ITU-T recommendation P.862 [23] in 2001. PESQ (Figure 4.1) is designed to analyze specific parameters of audio, including time warping, variable delays, transcoding, and noise.

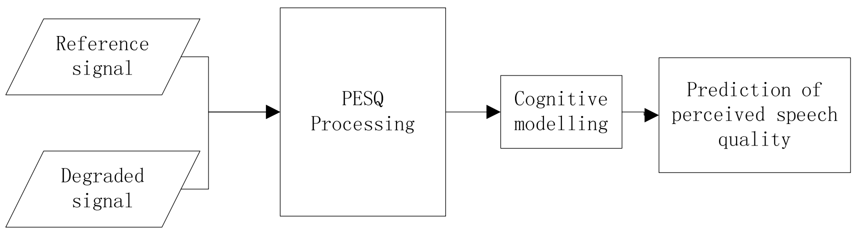


Figure 4.1 PESQ Block Diagram

PESQ score scales from -0.5 (worst) up to 4.5 (best), with a higher score indicating a better quality. In order to compare with the ITU MOS scale [29] ranges from 1.0 to 5.0, we transform the PESQ score to PESQ listening quality (PESQ-LQ) value, which is defined in ITU-T P.800 [29]. PESQ-LQ ranges from 1.0 to 4.5. 4.5 is usually the highest score obtained in a subjective test. Formula to transform the PESQ score (x) into the PESQ-LQ value (y) [21] is:

$$Y = \begin{cases} 1.0, & X <= 1.7 \\ -0.157268 * X^3 + 1.386609 * X^2 -2.504699 * X + 2.023345, & X > 1.7 \end{cases}$$

4.2 VQM

There are several different tools to measure the video quality. After some experiments and consideration, here we chose BVQM (Batch Video Quality Metric) as the measurement tool to analyze the video quality. In videoconferencing, the media data was transmitted by packets, but packet loss in this real-time communication is hard to avoid. Some frames of the video were lost, so we can’t use the frame-to-frame VQM tools, such as MSU VQMT. BVQM is a video quality metric software tool developed by the Institute for Telecommunication Sciences (ITS). It performs objective automated quality assessments of processed video clip batches.

BVQM compares the original video clip to the processed video clip and reports quality impairment on a scale from zero to one. The smaller the score is, the smaller the difference between the original and processed video is, and the better the video quality is.

4.3 Jperf

We did these experiments of QoE over a long distance on the Internet. The network situation can change at any moment. In order to be sure that all the results were generated in a similar network environment, we introduced Jperf to monitor the network situation.

Jperf is a free Java-based GUI tool (figure 4.2) that performs TCP and UDP packets measurements between two endpoints. By running Jperf on two computers as server and client over a network, data flows are sent between the computers and measurement results about the end-to-end bandwidth are returned.

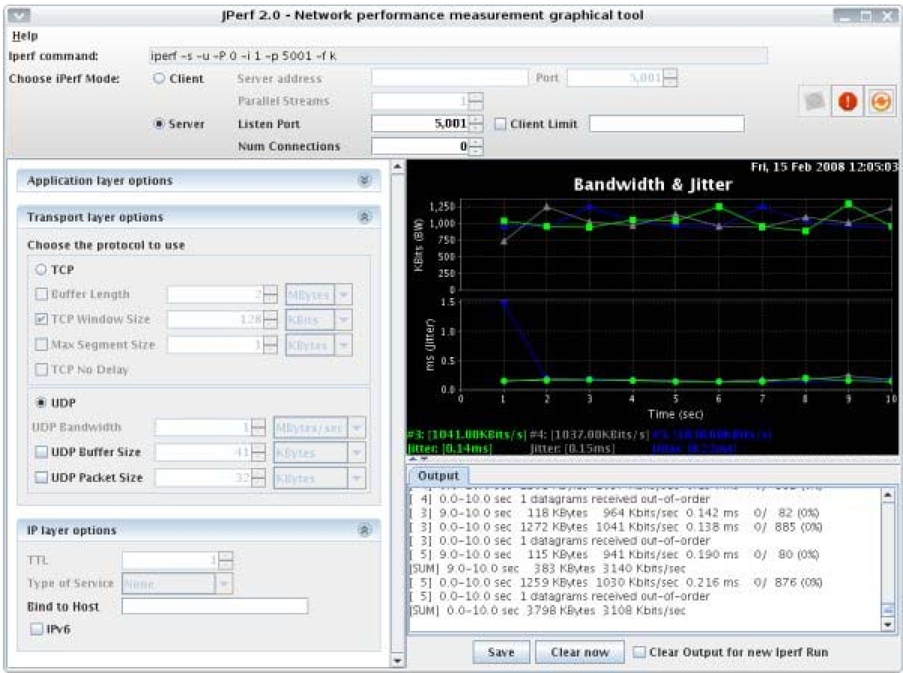


Figure 4.2 Jperf User interface

4.4 Audio and Video quality Experiments

4.4.1 Scenario description

There are four participants in this measurement. Two of them are located in the Netherlands; two of them are in China:

Client A: Jackstraw, 145.94.40.113, Delft, the Netherlands, Windows XP, with NAT and firewall, CPU: Inter(R) Core2 Duo CPU T7100 @1.80Ghz 1.79Ghz, 2.0GB RAM; 10/100 FastEthernet

Client B: Yue, 131.180.41.29; Delft, the Netherlands; Windows XP;
Inter(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz 2.99GHz; 3.25GB RAM; 10/100 FastEthernet;

Client C: LZH, 159.226.43.39, Beijing, China, Windows XP, with NAT and firewall
CPU: Inter core2 Duo CPU E8400 @ 3.00Ghz, Memory: 2.0GB, 10/100 FastEthernet

Client D: SFC, 124.228.71.177, HengYang, China, Windows XP, with NAT and firewall
CPU: Inter(R) Core2 Duo CPU T2250 @ 1.73Ghz 1.73Ghz, Memory: 2.87GB, ADSL 1MB

We use two original videos to measure: Video1 (Bit rate: 947kbps Size: 480x270 Frame rate: 24.94fps) and Video2 (Bit rate: 947kbps Size: 160x120 Frame rate: 10.00fps). According to the characteristic of the stream content in videoconference, the videos we broadcasted via virtual cameras are “TV news report”.

During the experiment, we captured at every participant the stream from the embedded multimedia player of each videoconferencing application with Camtasia Studio 6 and used wireshark to capture the traffic and Jperf to monitor network situation at each participant.

BVQM only supports uncompressed AVI and Big YUV video and the original and processed video must have the same frames. After the experiments, we needed to convert the video from other formats into the proper formats and use VirtualDub to cut all the sampled videos from the same beginning and the same length. At last, we input the calibrated original and received videos into BVQM tool to get the VQM score. In order to avoid the impact of the conversion, we kept the window size and the frame rate of all videos as same as the original video. Besides that, we used Camtasiz Studio 6 to capture the original video with different FPS from the original video, it turns out that the VQM score was zero. That means this way of capturing using Camtasia did not impact the result.

During the experiments, we used Jperf to monitor the network between the endpoints. The network situation is shown in Figure 4.2.

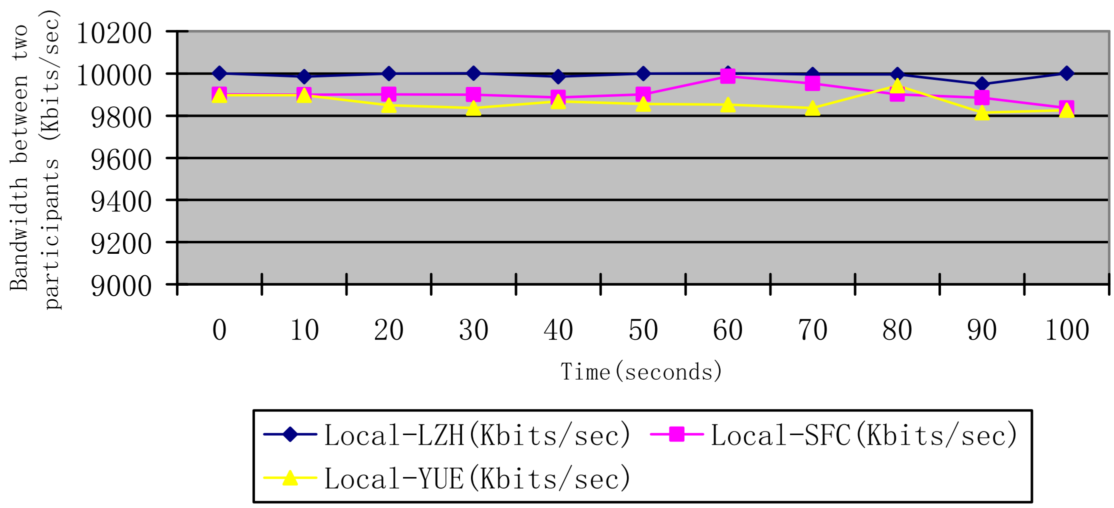


Figure 4.3 Network situations monitored by Jperf

From Figure 4.3, we can see that the network situations during our experiments are quite stable. Different applications work in a similar network environment.

4.4.2 PESQ results of audio quality

In these experiment, we captured the audio stream, and use PESQ to get the PESQ score. We then transform the PESQ score to the PESQ-LQ [21] value (referred to as PESQ_MOS in Table4.1). The audio quality result is given in Table 4.1.

REFERENCE	DEGRADED	PESQ_MOS
original.wav	Mebeam.wav	2.24
original.wav	Vsee.wav	3.08
original.wav	Nefsis.wav	3.15
original.wav	Qnext.wav	2.68

Table 4.1 PESQ_MOS of the audio quality

From the results in Table 4.1, we can see that the audio quality of Nefsis is the best, while Mebeam provides the worst audio quality.

4.4.3 Objective measurements results of video quality

In these experiments, VQM score described the video quality for different applications. The lower value of the VQM score, the better video quality is. The process and result of Video1 (Bit rate: 947kbps Size: 480x270 Frame rate: 24.94fps) are given below.

1. Mebeam

The original video image and the video snapshot displayed for four participants are shown in Figure 4.4 and Figure 4.5.

Figure 4.4 Original video snapshot

Figure 4.5 Snapshot of videos at four participants (from right to left: Jackstraw, Yuelu, Lzh, Sfc)

The corresponding VQM results are given in Figure 4.6

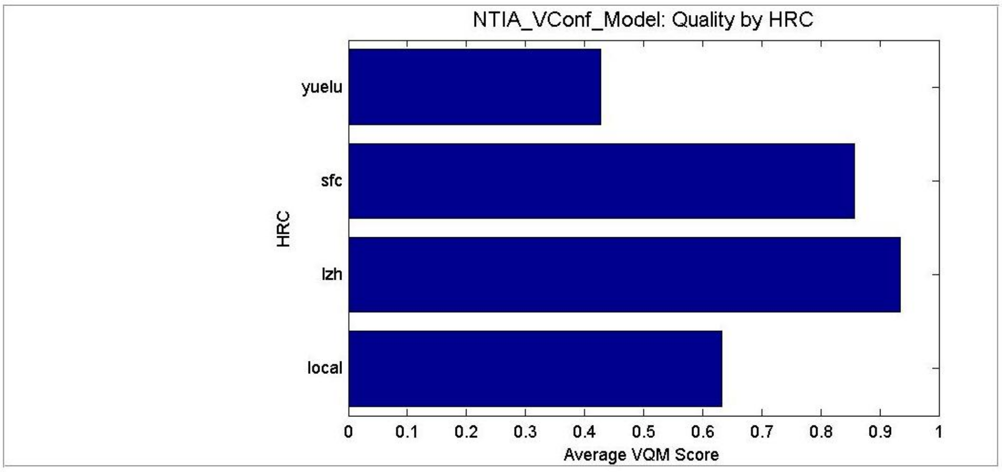


Figure 4.6 VQM result of Mebeam

Figure 4.6 shows the results generated by BVQM. We compared the original video and the processed video, which received by the host. The histogram described the VQM score between the original video and the processed video for different participants.

2. Qnext

The video snapshots displayed at four participants are shown in Figure 4.7.

Figure 4.7 Snapshot of videos at four participants for Mebeam

(From right to left: Jackstraw, Yuelu, Lzh, Sfc)

The corresponding VQM results are shown in Figure 4.8

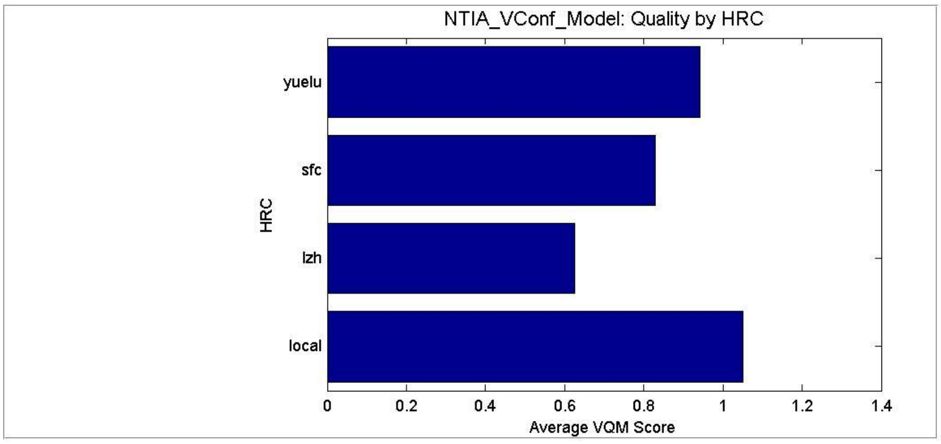


Figure 4.8 VQM result of Qnext

3. Vsee

The video snapshots displayed at four participants are shown in Figure 4.9.

Figure 4.9 Snapshot of videos at four participants for Vsee

(From right to left: Jackstraw, Yuelu, Lzh, Sfc)

The corresponding VQM results are shown in Figure 4.10.

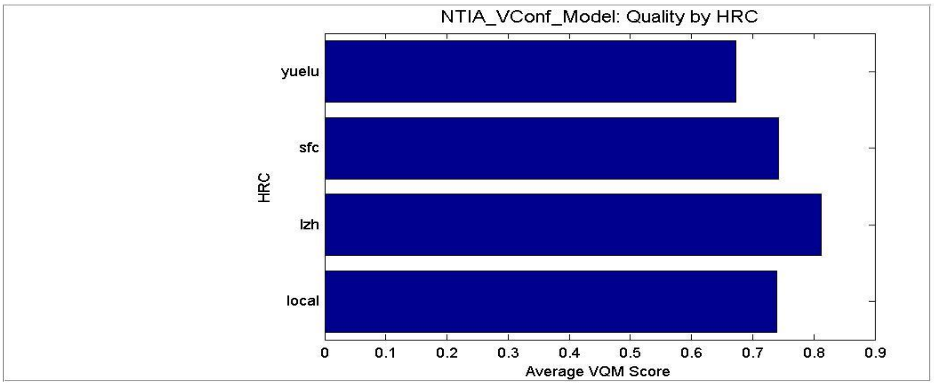


Figure 4.10 VQM result of Vsee

4. Nefsis

The video snapshots displayed at four participants are shown in Figure 4.11.

Figure 4.11 Snapshot of videos at four participants for Nefsis

(From right to left: Jackstraw, Yuelu, Lzh, Sfc)

The corresponding VQM results are shown in Figure 4.12.

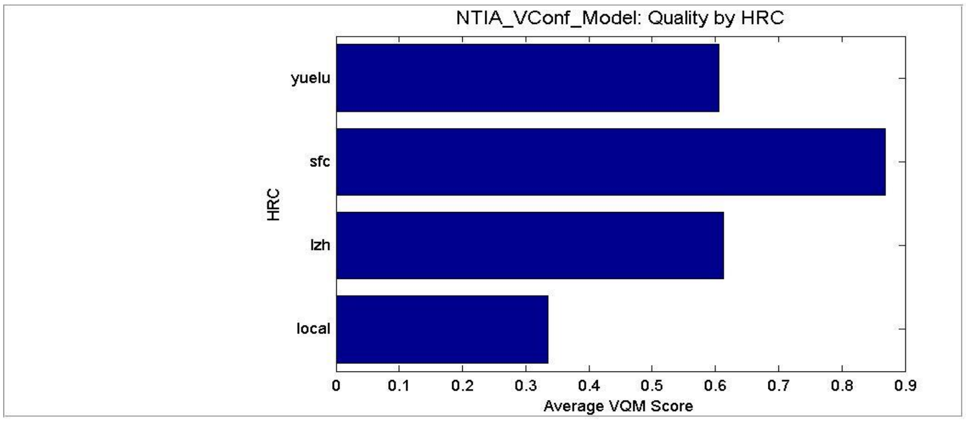


Figure 4.12 VQM result of Nefsis

To summarize, the overall results of the video quality at different participants for different applications when using Video1 can be seen in Figure 4.13.

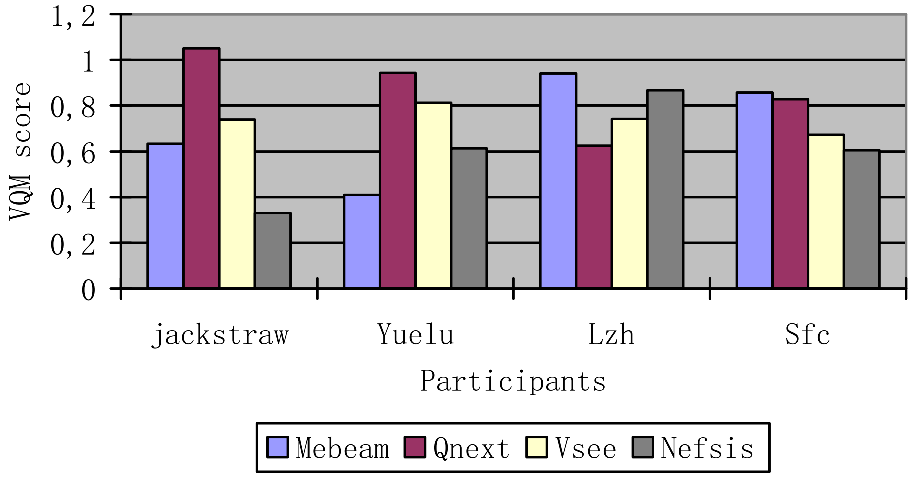


Figure 4.13 Overall result of the video quality when using Video1

Similarly, we also did the experiments of Video2 with different size and frame rate of Video1 (Size: 160x120, Frame rate: 10.00fps). The result of Video2 is in Figure 4.14.

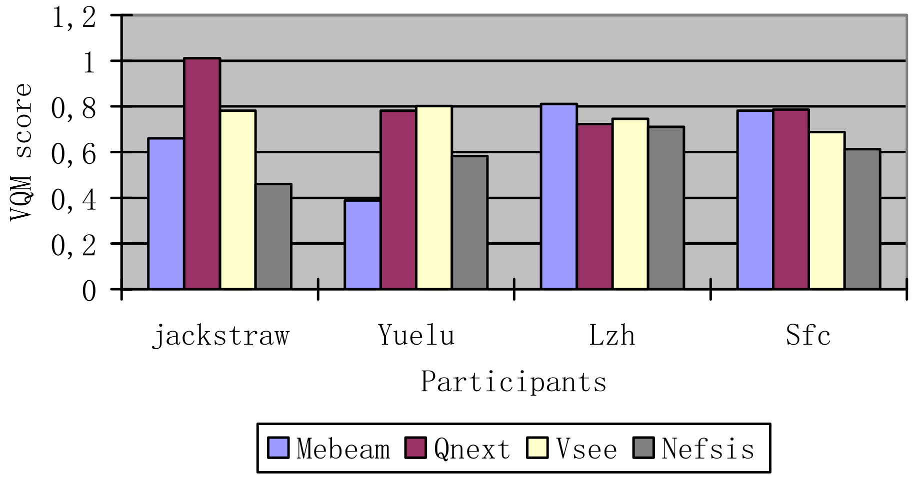


Figure 4.14 Overall result of the video quality when using Video2

We can see from Figure 4.13 and Figure 4.14 that Nefsis performs the best with an average VQM score of 0.53. Mebeam and Vsee have similar video quality, and their quality is approximate at 0.75 on average. Qnext didn't perform well, especially at the host side where the VQM score is up to 1.

Generally speaking, all desktop video conferencing applications cannot provide good quality between long distance conference participants.

In order to more clearly understand what level of video quality exactly the end user experienced, we selected 7 received video clips with different levels of VQM scores and the original video to do subjective measurements.

4.4.4 Subjective measurements of video quality

We asked 24 people to make a subjective assessment about the relation between VQM score and the MOS (Mean Opinion Score), which is a subjective call quality measurement perceived by the user. We gave these people 7 videos, which have VQM scores from 0.3 to 1.0. They watched the videos and gave MOS scores. MOS scales from 1 to 5. 1 represents “Bad”, 2 is “Poor”, 3 is “Fair”, 4 is “Good” and 5 is “Excellent”. Typically a score above 4.1 was considered to be very good, while a score under 2.5 was pretty lousy [17]. The video quality provided by a P2P TV application SopCast is around 4 [18].

Figure 4.15 shows the relation between the objective VQM score and the subjective MOS from 24 people.

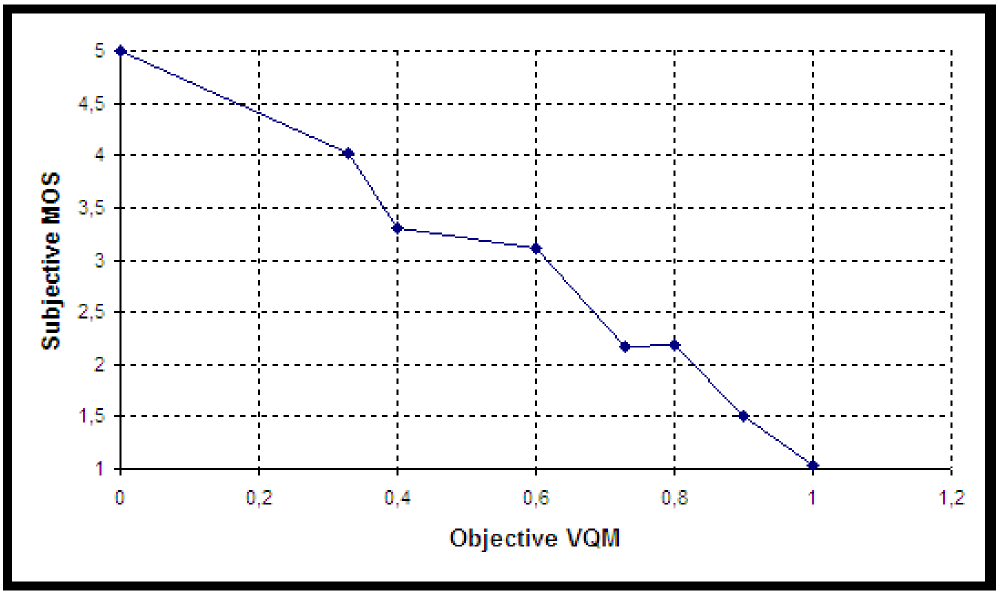


Figure 4.15 Relation between VQM score and MOS

From Figure 4.15, we can conclude a formula to express the relation between VQM score and MOS value, which is MOS = 5.3 – 4.1 * VQM.

In all, these desktop VC applications provide a video quality with average VQM score 0.74, which represents a MOS value around 2.3. The result shows that the video quality of DVC is “Poor”.

4.5 Interactivity

Interactivity is also called synchronization among participants or peer lag. When we are in the video conference, our interconnection might be not synchronized sometimes. We try to measure the different audio delays and video delays between any 2 participants with qualitative experiments in this section.

4.5.1 Scenario description

1. Four participants: two are in the Netherlands, two are in China, which is the same as the VQM experiments.
2. The standard Internet time [27] for everyone is the same.
3. "Dual-tone multi-frequency" Audio.
4. Jperf to monitor the network situation.
5. Camtasia Studio 6 to capture the video at each participant.

4.5.2 Experiment processes and results for audio delay

We measure the different audio delays between any two participants for all applications. We measured the audio delays among participants by injecting in the video an artificial DTMF (Dual-tone multi-frequency) tone. We sent and recorded the audio at Client A. Other participants kept their speaker and microphone on, but did not produce extra audio. Based on the recorded audio tracks, we compared the time difference of the audio marker was sent from Client A and the time the same audio marker was heard again at Client A after the transmitted audio was played, recorded, and retransmitted by a client. The time difference is approximately twice the one-way audio delay plus the processing delay at a client. The process of the method is shown in Figure 4.16.

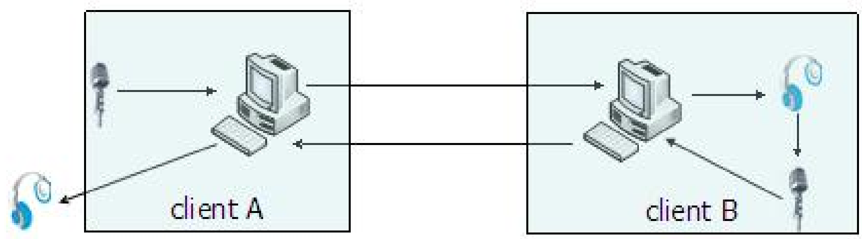


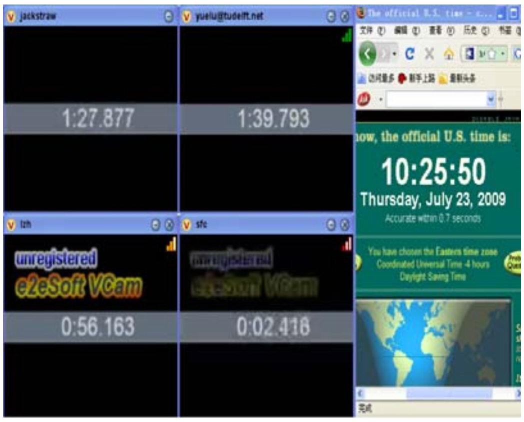
Figure 4.16 Process of the audio delay

Our results revealed 1 s, 1.4 s, 0.2 s and almost 0 s on average for Mebeam, Qnext, Vsee and Nefsis respectively. Qnext in this case provided the largest audio delay among the users. Nefsis performs the best.

4.5.3 Experiment processes and results for video delay

We captured the video of local, LZH, SFC in different videoconference applications. In each video, there are a standard time and the video as a counter in videoconference. We have cut them with VirtualDub to compare the delay. We chose a same time shown in the standard time as start, and then compared the time shown in the counter. We took Vsee and participants Jackstraw and Lzh for example, these are the video images in Jackstraw and Lzh:

Jackstraw



Lzh

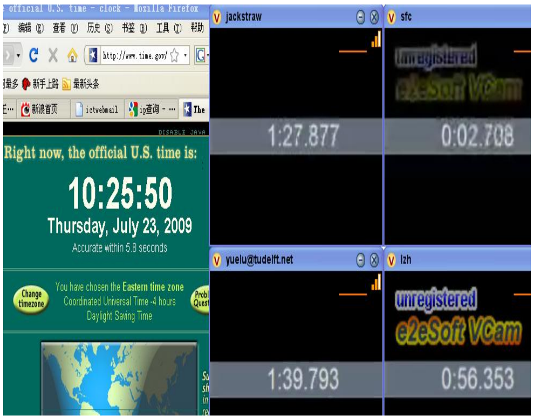


Figure 4.17 Example of Interactivity measurements

From the images above, we can get the delay between jackstraw and lzh, the corresponding parts of the Figure are shown italic and bold in Table 4.2.

	Jackstraw		Yue		Lzh		Sfc	
	10:25:50 AND 10:25:53							
Jackstraw	***1:27.877***	1:31.084	***1:39.793***	1:43.392	***0:56.163***	0:59.755	***0:02.418***	0:06.100
Yue	1:28.084	1:31.084	1:40.000	1:42.996	0:56.249	0:59.357	0:02.708	0:05.708
Lzh	***1:27.877***	1:30.878	***1:39.793***	1:42.787	***0:56.353***	0:59.357	***0:02.708***	0:05.829
Sfc	1:27.739	1:31.146	1:40.043	1:43.362	0:56.726	1:00.028	0:02.820	0:05.211

Table 4.2 Recorded time in the example Interactivity measurement

Each row in the table describes the times in one participant at two different time, and columns describe the times of different participants shown on one participant's screen at one time. Still took Jackstraw and Lzh as example,

At 10:25:50,

The delay from jackstraw to lzh is: D_1 = |1:27.877 - 1:27.877|

The delay from lzh to jackstraw is: d_1 = |0:56.353 - 0:56.163|

And at another time 10:25:53,

The delay from jackstraw to lzh is: D_2 = |1:31.084 -1:30.878|

The delay from lzh to jackstraw is: d_2 = |0:59.755 – 0:59.357|

Delay (from jackstraw to lzh) = $(D_1 + D_2)/2$ = 0.103

Delay (from lzh to jackstraw) = $(d_1 + d_2)/2$ = 0.25

$$\text{Delay} = \sum_{i}^{N} \frac{D_i}{n}$$

Similarly, we made the experiments for all applications,

1. Mebeam

The 2 sample time are at 12:04:15 and 12:04:12

	Jackstraw		Yue		Lzh		Sfc	
12:04:15 AND 12:04:12								
Jackstraw	0:11.605	0:08.516	0:05.518	0:01.119	0:15.815	0:10.609	0:20.920	0:15.920
Yue	0:07.032	0:04.011	0:07.032	0:04.011	0:15.119	0:12.129	0:20.741	0:15.920
Lzh	0:09.923	0:05.208	0:05.518	0:01.119	0:17.623	0:14.632	0:20.920	0:15.920
Sfc	0:09.923	0:05.023	0:06.100	0:02.708	0:15.815	1:12.226	0:23.734	0:20.424

Table 4.3 Recorded time in the Interactivity measurements for Mebeam

The result can be plot as follows:

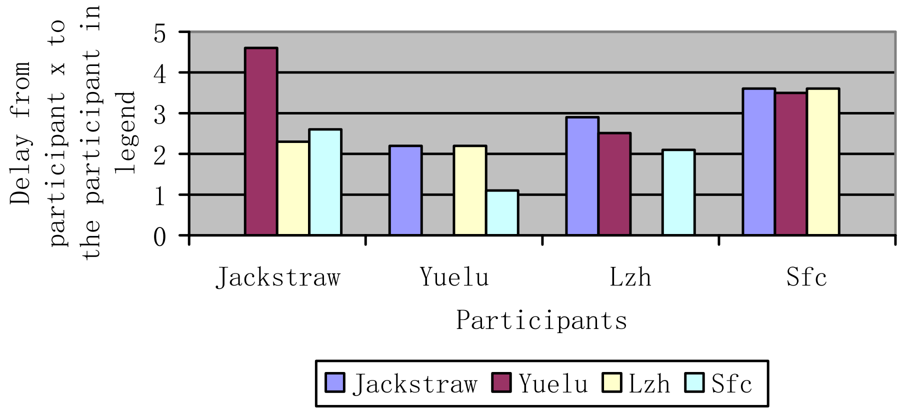


Figure 4.18 Result of Mebeam's Interactivity between different participants

The Figure 4.18 shows that the conversation delay in Mebeam is over 2.8 second. The synchronization among participants in Mebeam performs not well.

2. Nefsis

The 2 sample time are at 12:00:00 and 12:00:03

	Jackstraw		Yue		Lzh		Sfc	
12:00:00 AND 12:00:03								
Jackstraw	1:33.092	1:35.896	0:24.128	0:26.932	0:20.527	0:23.339	0:03.697	0:06.413
Yue	1:32.886	1:35.691	0:24.317	0:27.242	0:20.920	0:23.339	0:03.697	0:06.413
Lzh	1:32.877	1:35.878	0:24.793	0:27.787	0:20.353	0:23.357	0:02.708	0:05.829
Sfc	1:32.739	1:35.146	0:24.043	0:27.362	0:20.726	0:23.028	0:02.820	0:05.211

Table 4.4 Recorded time in the Interactivity measurements for Nefsis

The result can be described as:

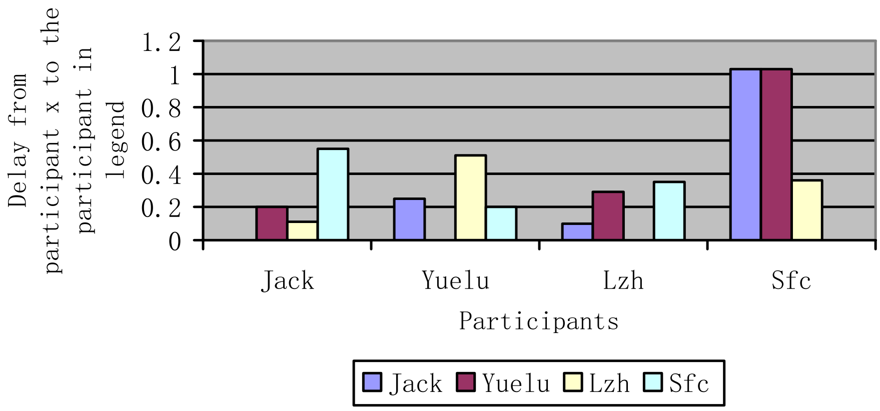


Figure 4.19 Result of Nefsis's Interactivity between different participants

It shows that most of the participants' conversation delay in Nefsis is less than 500 ms, the average delay time is approximate 600 ms. That means the synchronization among peers in Nefsis is good.

3. Vsee

The 2 sample time are at 10:25:50 and 10:25:53

	jackstraw		Yue		Lzh		Sfc	
10:25:50 AND 10:25:53								
jackstraw	1:27.877	1:31.084	1:39.793	1:43.392	0:56.163	0:59.755	0:02.418	0:06.100
Yue	1:28.084	1:31.084	1:40.000	1:42.996	0:56.249	0:59.357	0:02.708	0:05.708
Lzh	1:27.877	1:30.878	1:39.793	1:42.787	0:56.353	0:59.357	0:02.708	0:05.829
Sfc	1:27.739	1:31.146	1:40.043	1:43.362	0:56.726	1:00.028	0:02.820	0:05.211

Table 4.5 Recorded time in the Interactivity measurements for Vsee

The result can be described as:

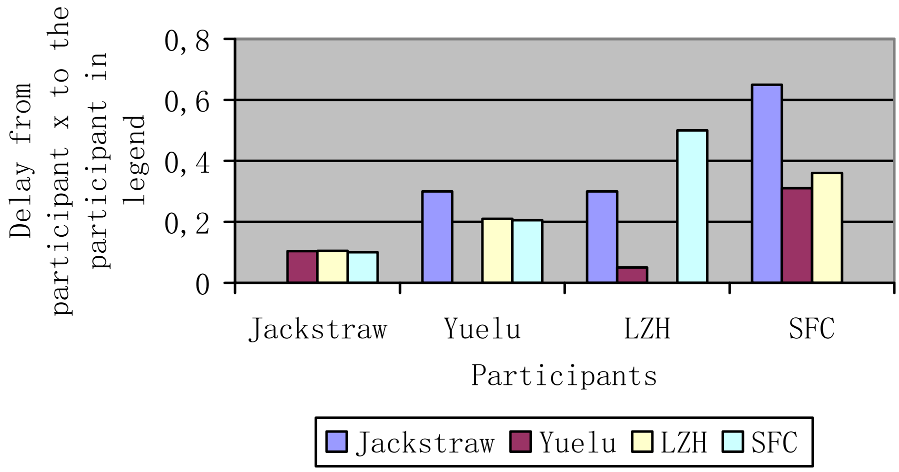


Figure 4.20 Result of Vsee's Interactivity between different participants

The time synchronization in Vsee is good. Most of the participants' delay is under 500 ms.

4. Qnext

The 2 sample time are at 11:50:05 and 11:50:10

	Jackstraw		Yue		Lzh		Sfc	
	11:50:05	AND	11:50:10					
Jackstraw	0:28.818	0:33.926	0:15.423	0:20.320	0:07.712	0:12.704	0:09.011	0:13.923
Yue	0:28.942	0:34.132	0:15.423	0:20.527	0:07.712	0:12.930	0:09.114	0:14.217
Lzh	0:28.942	0:33.926	0:15.423	0:20.424	0.08.000	0:12.930	0:09.001	0:14.032
Sfc	0:29.042	0:33.996	0:15.608	0:20.824	0.07.800	0:13.030	0:09.118	0:14.306

Table 4.5 Record time of Qnext's Interactivity measurement

The result can be described as:

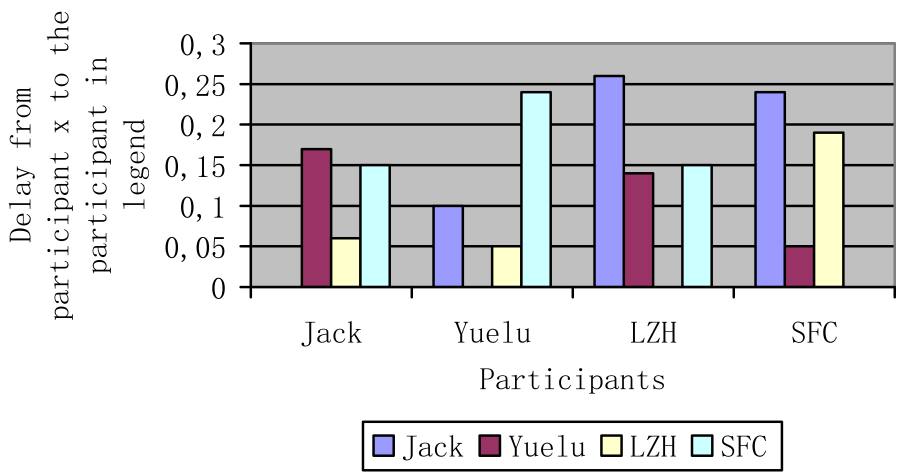


Figure 4.21 Result of Qnext's Interactivity in different participants

It shows that most of the participants' conversation delay among participants is less than 250 ms. The synchronization among peers in Qnext is very good.

From the results of the interactivity measurements, we can see the video delay among different participants of Mebeam is over 2.8 seconds. Other desktop VC applications are with an average delay about 500 ms. The reason of the large conversation delay of Mebeam is that Mebeam is a web browser-based application. Brower-based application don't have the client software to perform the calculations [28], they rely on the servers to do the processing. So the users must suffer the latency of sending the information over the Internet to the remote server and receiving the updated information. Moreover, all users' processing in the server will cause further delays.

In our experiments, the standard Internet time has accuracy with 0.4-0.8 seconds. And in IP videoconferencing scenarios the maximum communication delay recommended by ITU [24] is 400 ms. Considering the accuracy of the standard Internet time, the maximum conversation delay should be less than 1200ms then. Therefore, we can conclude that Mebeam performs not well in terms of the video interactivity, and other three applications are very good.

4.6 Audio-Video Synchronization

Audio-video synchronization refers to the relative timing of sound and image of a streaming content.

According to a research of Blakowski and Steinmetz in 1996 [25], the spoken woed before seeing the lips move is more unnatural to a viewer. In 1998, International Telecommunications

Union recommendation published BT.1359, stating the relative timing of sound and vision for broadcasting. ITU and some other researches [26] have suggested that thresholds of timing for viewer detection are about +45 ms to -125 ms, and the thresholds of acceptability are about +90 ms to -185 ms.

In this group of experiments, we decided to analyze the A/V synchronization provided by each video conferencing application with an "artificially generated" video test sample, which is 25FPS and provided by the TNO. In the test sample, some markers were inserted in the audio component and video component. Every three seconds, a red full screen image appears in the video while an audio waveform "beep" was produced. The video and audio markers are synchronized to transition from one state to another at the same time. Figure 4.21 shows the video frames. Figure 4.22 shows the audio clips.

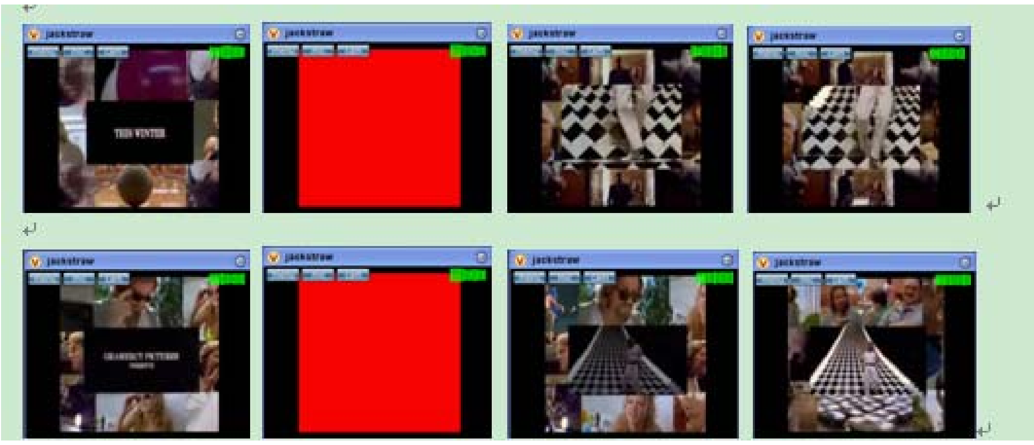


Figure 4.22 video frames inA/V synchronization measurement

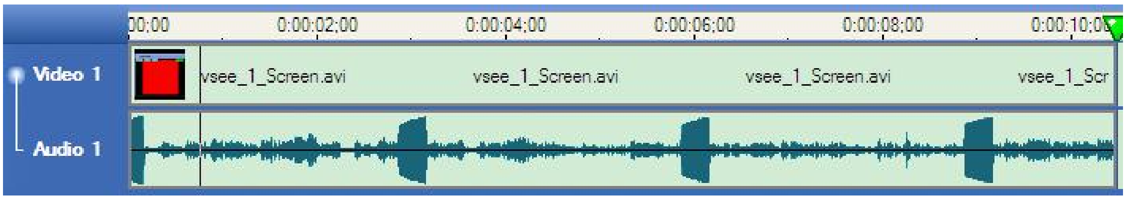


Figure 4.23 audio clips of A/V synchronization

Similar to the experiments of testing the video quality, at each participant we captured the videos from all other participants. After that, the audio and video tracks were extracted and compared offline. An average lag between the video and audio is 650 ms for Mebeam, 470 ms for Qnext, 400ms for Vsee and 350ms for Nefsis. Obviously, the A/V lags are over the suggestion of timing thresholds suggested by ITU [26]. The reason of the large lag between audio and video in videoconferencing service is:

1. The low video quality because of a large amount of frame losses. In our experiment, some red full screen image marker frame loss caused the lagging between audio and video.
2. The transmission delay. For instance, the delays at the transmitter, in the network and at the receiver. These delays include the capture delay, encoding delay, packetization and play out delay at the endpoint hardware devices, gateways and transcoders delay, the decoder delay etc.

4.7 Worst case study

4.7.1 Global experiment under unstable conditions

We did a group of experiments on a Saturday evening (in June, 2009) in Chinese time. There are four participants, two of them are located in the Netherlands; two of them are in China.

From our Jperf plots (see Figure 4.23), we can see that the connection between the host and two of the participants in China is really unstable. Under this situation, the two participants in China cannot launch the videoconferencing by Mebeam, Nefsis and Qnext. Vsee can still work, but the quality is not good. The VQM score is given in Fugure 4.24.

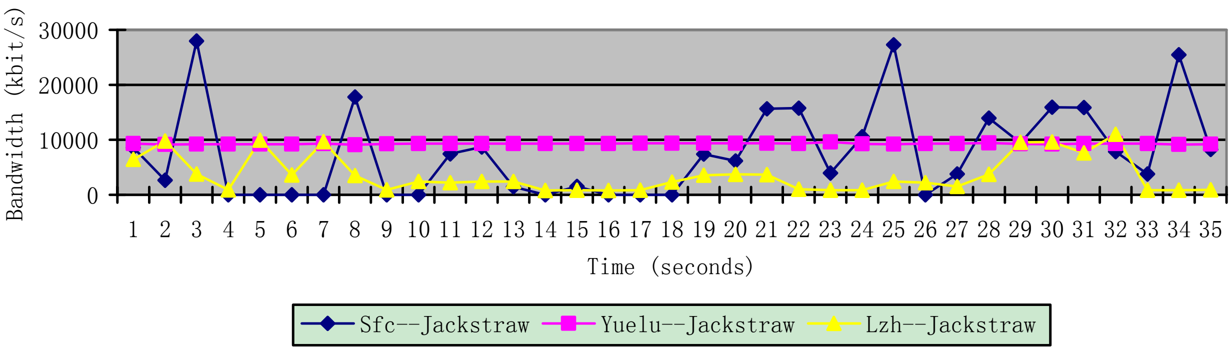


Figure 4.24 Jperf plots under the unstable situation

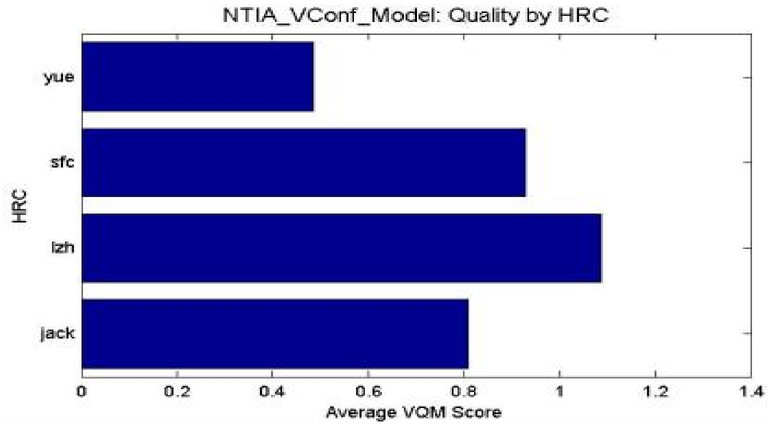


Figure 4.25 VQM score of Vsee under the unstable situation

From Figure 4.23, we can see that the network situation between Yue and Jack is fine, so the video quality of these two is not too bad (Yue is 0.4858 and Jack is 0.8077, which can be seen in Figure 4.24). However, quality of the video from the other two participants is really bad (Lzh is 1.0874 and Sfc is 0.9287).

4.7.2 Min Upload bandwidth to launch a conference

We repeated experiments adjusting the upload rate upper bound (using Netlimiter 2 Pro [19]) at each participant for a particular video conferencing application to test the user's upload bandwidth minimally required to launch a video conference.

For Qnext, the threshold is 50 Kbit/s. That means, if an end user's available upload bandwidth is lower than 50 Kbit/s, he cannot launch a video conference with other participants. For Vsee, the threshold is also 50Kbit/s; for Nefsis it is 30Kbit/s; and for Mebeam it is 7 Kbit/s.

4.8 Other parameters related to QoE

From our experiments, we found that the CPU, memory size and the environment are also important factors, which can affect the QoE of the Desktop VC. When the CPU or memory is overloaded, the Desktop VC can't work stably. In Desktop VC, the processing of a video stream must be in real-time, which requires a lot of resources. If the CPU or memory is overloaded and can't deal with the resource-consuming multipoint Desktop VC in time, the QoE of Desktop VC will be affected a lot. Besides that, the environment will affect the QoE too. Different environment, such as light, noise, obviously will influence the user's perspective.

4.9 Discussion

Through our experiments described this section, we have learned more about the Quality of Experience of different desktop multipoint videoconference applications. In general, we can observe that:

1. Desktop VC applications have an average VQM score of 0.74, which represents a MOS value around 2.2. Therefore, we can conclude that over long distances, desktop VC applications cannot provide good video quality.
2. Interactivity of Mebeam performs not well. The delay among endpoints is over 2.8 second. For other desktop VC applications, the interactivity is much better. The delays among endpoints are below 500 ms.
3. Because of frame loss during the transmission, the A/V Synchronization is not good, with lags between 350 ms and 650 ms, which is over thresholds of timing suggested by ITU.
4. Considering the audio delay, video delay and the A/V synchronization results, we can conclude that the delay introduced by the application when uploading is large for Qnext.
5. In Desktop VC applications, centralized P2P VC architecture, such as Qnext, is not a good choice to provide VC service. The amount of the I/O traffic at the host (super node) is too big, which causes the extremely bad video quality at the host.
6. Besides bandwidth, network stability, and stream codec, CPU and memory are also parameters to affect QoE in Desktop VC together with bandwidth, network stability, and stream codec.
7. All desktop VC applications require a minimum upload bandwidth ranging from 7 Kbit/s to 50 Kbit/s to establish the conference.
8. According to our experiments, the video quality of Desktop VC is sensitive to the bandwidth and network stability. An effective solution should be to find a way to balance the trafficload and the quality.

5

Conclusions

There are hundreds of multipoint videoconference applications on the Internet, more and more new technologies and applications were developed and provided to the customers. However, few QoE measurements of these VC systems have been referenced.

In this thesis, we made a survey of existing multipoint videoconference technologies and applications. We chose four different typical desktop videoconference systems representing four different architectures for further study. We did several sets of measurement experiments to understand the behavior and mechanism of these systems. From the process and result of these experiments, we got the details of the login process, call establishment process, I/O traffic, packets distribution, transfer protocols, and the topology of different applications.

We also designed sets of experiments to measure different aspects of QoE. Audio quality, video quality, Interactivity (audio delay and video delay), Audio-Video Synchronization and worst cases are studied.

The results of all the experiments show that Nefsis performs the best, no matter in the aspect of video quality or interactivity, or even in audio-video synchronization. Vsee performs well in most of time. In Qnext, because of the heavy I/O traffic, the conference host cannot perform well when the number of participants in the conference is more than three. Other participants in Qnext got better QoE than the host. Mebeam cannot be stable all the time. When it is stable, its quality is acceptable. When the network bandwidth is limited or network status is unstable, there is no desktop videoconference application able to work properly.

Based on these outcomes, we can know how these different VC systems work and what should be paid attention to when new VC applications need to be designed and developed in the future. According to the study of all these VC applications, we have some suggestions on the desktop videoconference on the internet:

1. The network situation including network bandwidth and network stability are the most important parameters in Desktop VC. In order to adjust the traffic load generated by them with different network status, we suggest the VC communication should be divided into several conditions. For example, we can set low bandwidth; medium bandwidth and high bandwidth corresponding to the different video codecs. In this way, the best possible quality can be achieved with adapting the traffic.
2. In P2P Desktop VC applications, both centralized P2P VC applications (Qnext) and full-mesh P2P VC applications (Vsee) have obvious disadvantages. The centralized P2P VC application has too heavy traffic load at the host, which makes the QoE at the host very bad. Full-mesh P2P VC applications have heavier traffic load at each participant than centralized P2P VC application (except the host in Qnext).
3. Web browser-based VC applications is easy to use and don't need to install anything. However, due to the overhead processing time at the server, the conversation delay is huge compared with the VC applications with local stream processing at clients.
4. We should find an approach to integrate the P2P and S/C VC architectures. Nefsis made a good start. It has the client software to process the media data, and also has many servers located all over the world as access points to connect different participants.
5. During the conference, it is better to consider the different priority of the audio and video quality in different scenarios. For example, the audio quality should be guaranteed with a higher priority in remote education service, while in remote healthcare the video quality should get higher priority.
6. Make sure someone who leaves the conference by accident could join the meeting again.

6

Future work

Desktop VC developed very fast in recent years, lots of new steaming technologies and network architectures appeared. More detailed measurement experiments are necessary to better understand the multipoint Desktop VC systems. The following research can be the future work:

1. Measure other P2P structure Multipoint Desktop VC systems compared with our results. Such as tree structure (i.e. Vmukti).
2. Measure the impact of CPU and memory to a DVC application. We can use passive measurement to monitor the CPU and RAM changes in different VC operations. We can also use active measurement to change the CPU and RAM to see how is the quality of experience changed.
3. Design some measurement experiments to test the security of VC. For example, we can analyze the performance in different security strategy.
4. Quantificationally measure how bandwidth or background traffic flows affect QoE.
5. According to the results of this thesis, it is possible to design and implement better Desktop VC applications, which can provide good QoE and support more simultaneous participants.

Reference

[1] Skype Limited, http://www.skype.com

[2] Salman A.Baset and Henning Schulzrinne, "An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol", Department of Computer Science. Columbia University, New York, USA, 2004

[3] M. Reha Civanlar, Oznur Ozkasap, Tahir Celebi, "Peer-to-Peer multipoint videoconferencing on the Internet", Signal Processing. Image Communication, 2005vol.20(no.8)

[4] K.Salah, "An Analytic Approach for Deploying Desktop Videoconference", Department of Information and Computer Science King Fahd University of Petroleum and Minerals, Saudi Arabia, 2006

[5] Richard Spiers and Neco Ventura, "An Evaluation of Architectures for IMS Based Video Conferencing", University of Cape Town, Rondebosch South Africa, 2009

[6] H. Horiuchi, N.Wakamiya, and Murata, "A network construction method for a scalable P2P video conferencing system", Chamonix, France, 2007

[7] Yubing Wang, Mark Claypool, and Robert Kinicki, "Impact of Reference Distance for Motion Compensation Prediction on Video Quality", San Jose, California, 2007

[8] Ira M. Weinstein, "Making the Best of ISDN-Based Videoconferencing", Wainhouse Research, 2004

[9] The International Telecommunication Union (ITU), ISO/IEC JTC1/SC29/WG11 N3536, Beijing, 2000

[10] The International Telecommunication Union (ITU) E 10669 (11/98) Recommendation H.323

[11] Scott Firestone, Thiya Ramalingam, and Steve Fry, "Voice and Video Conferencing Fundamentals", Cisco Press, Indianapolis, USA, 2007

[12] Network Working Group, RFC 3261 AT&T, 2002

[13] http://en.wikipedia.org/wiki/Videoconferencing

[14] http://www.mebeam.com

[15] Nefsis Corporation, San Diego, CA, http://www.nefsis.com

[16] Qnext Corporation, USA, http://www.qnext.com/

[17] Vsee Corporation, USA, http://www.vsee.com

[20] B.Fallica, Y. Lu, F.A. Kuipers, R. Kooij, and P. Van Mieghem, "On the Quality of Experience of SopCast", in Proc. of the First IEEE International Workshop on Future Multimedia Networking (IEEE FMN´08), Cardiff, Wales, September 18, 2008.

[21] A. W. Rix. “A new PESQ-LQ scale to assist cocmparison between P.862 PESQ score and subjective MOS”, ITU-T SG12 COM12-D86, May 2003

[22] Network Working Group, RFC 3261 AT&T, 2002

[23] The International Telecommunication Union (ITU) (02/01) Recommendation P.862

[24] Ivano Bartoli, Giovanni Iacovoni, Fabio Ubaldi, “A synchronization control scheme for Videoconferencing services”, Journal of multimedia, Vol.2, No.4, August.2007

[25] Blakowski, G., Steinmetz, R., “A media synchronization survey: reference model, specification, and case studies”, Selected Areas in Communications, IEEE Journal on, Volume: 14, Issue: 1, Jan.1996

[26] Joseph L. Lias. “HDMI's Lip Sync and audio-video synchronization for broadcast and home video”, Simplay Labs, LLC, August.2008

[27] http://www.time.gov/timezone.cgi?Eastern/d/-5/java

[28] Mark S. Silver, “Browser-based applications: popular but flawed?”, ISeB (2006) 4: 361–393 DOI 10.1007/s10257-005-0024-3, 16 February 2006

[29] The International Telecommunication Union (ITU) (08/96) Recommendation P.800