Escola Politècnica Superior de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA

INSTITUT FÜR NACHRICHTENTECHNIK UND HOCHFREQUENZTECHNIK

TREBALL DE FI DE CARRERA

Title: Netmeeting: Performance and optimisation for UMTS network

Author: Eduardo Dijort Romagosa

Professor: Markus Rupp

Advisor: Olívia Némethová

Data: 25 de febrer de 2005

Title: Netmeeting: Performance and optimisation for UMTS networks

Author: Eduardo Dijort Romagosa

Professor: Markus Rupp

Advisor: Olívia Némethová

Data: 25 de febrer de 2005

Summary

Remote office applications like Netmeeting are very popular especially in big companies spread over different locations. Such applications are true multimedia applications allowing simultaneous speech, video conferencing, screen or document sharing, chatting and sharing the blackboard. Intention of this diplom thesis is to make a feasibility study for Netmeeting over wireless networks and based on that to propose the suitable network parameters.

The project has following parts:

· Research the types of media and their compression used/possible for Netmeeting

· Investigation of the stream produced by Netmeeting (packetization)

· Investigation of user requirements on quality of service parameters for particular media and their synchronization in Netmeeting by means of testing with Dummynet

· Providing a set of key performance parameters for Netmeeting

· Assessment of the related UMTS network parameters and their settings

· Conclusion on feasibility of Netmeeting over UMTS

Titulo: Netmeeting: Performance and optimisation for UMTS networks

Autor: Eduardo Dijort Romagosa

Director: Markus Rupp

Supervisora: Olívia Némethová

Fecha: 25 de febrer de 2005

Resumen

Las applications remotas tipo Netmeeting son muy populares especialmente en las grandes compañias que disponen de grandes oficinas repartidas en diferentes localizaciones, muchas veces en diferentes partes del mundo. Las verdaderas aplicaciónes multimedia simultean voz y datos, como pueden ser voz, video conferencia, compartir escritorio, compartir documentos, mensajes instantaneos. La intención de este proyecto es la de realizar un estudio sobre el tipo de tráfico sobre redes moviles and sobre esto proponer unos parametros optimos para el tipo de red.

El proyecto está repartido de la siguiente forma

Investigar los tipos de protocols, medios, compression que NetMeeting puede utilizar

· Analizar el tipo de stream que es producido por Netmeeting (paquetización.)

·Analizar los requerimentos de calidad de servicio que son necesarios por cada uno de los servicios ofrecidos de forma objetiva usando el programa DummNet para simulat diferentes entornor.

· Obtener un conjunto de llaves para optimizar los parametros de NetMeeting

· Seleccionar y ajustar los parametros relacionados con las redes UMTS y los ajustes que se pudieran llevar a cabo.

· Se popondrá unas conclusiones sobre cada servicios y se popondrá los diferentes RAB.

Index

Introduction 7
Chapter 1. NetMeeting characteristics 8
1.1 Introducction..8
1.2 Application introduction 8
1.2.1 Speech...9
1.2.2 Video...10
1.2.3 Desktop sharing 10
1.2.4 Chat 12
1.2.5 Whiteboard. 12
1.2.6 Transfer a file 13
1.3 Protocol Stack 14
1.3.1 T.120..14
1.3.2 T.121...15
1.3.3 T.122..15
1.3.4 T.123..16
1.3.5 T.124..16
1.3.6 T.125..16
1.3.7 T.126..17
1.3.8 T.127..17
1.3.9 T.128..17
Chapter 2. Scenario 18
2.1 Packetization of H.263 18
2.2 Speech 21
2.3 Desktop sharing 23
2.4 Transfer File 26
2.5 Whiteboard...28
2.6 Chat...30
Chapter 3. MOS measurements. 31
3.1 Programs...31
3.1.1 Network Address Translator (NAT)...31
3.1.2 Ethereal 32
3.1.3 Dummynet 35
3.1.4 Extract.exe 36
3.2 Measurements description 38
3.2.1 Desktop sharing 38
3.2.2 Chat 40
3.2.3 Whiteboard. 42
3.2.4 Transfer a file 43
3.3 NetMeeting: Subjective quality evaluation 43
Chapter 4. UMTS network 58
4.1 UMTS channels...58
4.1.1 The Medium Access Control Protocol..59
4.1.1.1 MAC functions 59
4.2 Radio link Architecture 60

4.3 RLC Functions.........61
4.4 The Packet Data Convergence Protocol.........61
4.5 QoS in UMTS.........61
4.6 Radio Access Bearers (RABs) for UMTS.........62
4.6.1 Calcul of maximum number of bits per TTI.........66
4.6.2 Video.........66
4.6.3 Audio.........68
4.6.4 WhiteBoard.........70
4.6.5 Transfer File.........72
4.6.6. Chat.........74
4.6.7 Desktop Sharing.........76
Chapter 5. Conclusions.........78
Chapter 6. Abbreviations.........82
Chapter 7. Bibliography.........84
APPENDIX.........88
APPENDIX A Code of the program.........88

Introduction

This document is a recommendation about how the services provided by NetMeeting application can be transmitted and how to use this knowledge to utilize the network resources of UMTS efficiently.

In UMTS networks there are defined different classes of service with different quality of service requirements. This for NetMeeting is perfect that each service needs a different type of traffic (different bandwidth, delay, jitter).

To do it, I will analyse the different type of traffic for each service generated. It will be analysed the different packet and delay of the service. All of this is making an analyse with various tools, and creating a program to obtain the information that has been extracted of the network.

To have an subjective evaluation of each service I made different measurements for different services (MOS). For every service different delay and different bandwidth applies that is possible to have in UMTS networks.

UMTS networks offer a QoS. There are 4 different class how is possible to read in this document (in the correspondent chapter). At the end of the work different RAB that would be used to transmit the different traffic that has been generated by NetMeeting are proposed.

Chapter 1. NetMeeting characteristics

1.1 Introducction

NetMeeting is an application that brings a new form that the people can communicate. With NetMeeting the people can use the computer to have a video conference, speech connection, chat discussion, application sharing and other services offered by NetMeeting.

NetMeeting is a tool that allows a real-time communication between one or more colleagues. The services that one can share with his partners are: speech, video, chat, file transfer, whiteboard and desktop sharing.

1.2 Application introduction

To use the program, first step is to start a communication with one computer. To establish the communication the user has to know the IP public or the name of the computer and if it is present within the same Local Area Network (LAN) and private IP. There is also possibility to connect via a NetMeeting server, used to connect the computers that are not connected to a common LAN. In the Figure 1-1 the NetMeeting application appearance can be seen. The program is easy to use.

Each function has a number according to the Figure 1-1. Each number is a different function that NetMeeting can perform:

1. Dial an computer according to its IP or computer name. This allows starting a communication between with a colleague.

2. Hang up the communication. Stops the communication.

3. LDAP service is a standard to find persons in a database if NetMeeting server has to be used.

4. Program sharing allows to share a program in a remote computer

5. Chat is a service that allows to exchange text messages between two computers

6. Whiteboard is an application that allows to share the Paint application between the communicating persons.

7. File transfer allows to transfer a file from one computer to another.

8. IP or computer name input box. The name of the computer the communication has to be established with it.

9. Web cam screen allows to see the face of the other person in the conference.

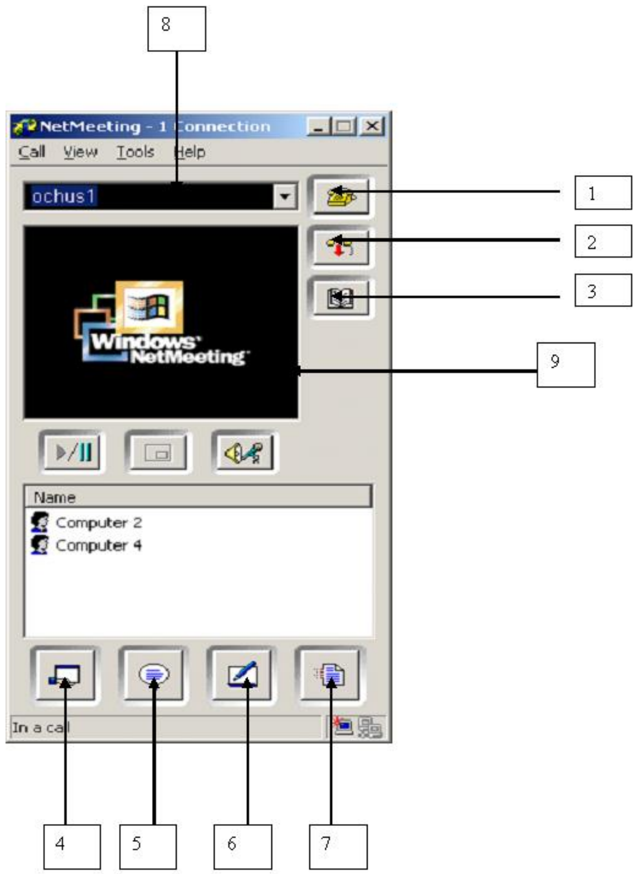

Figure 1-1 NetMeeting Program

NetMeeting has been created like an industry standard. All the different layers are based on standard.

1.2.1 Speech

The speech communication begins in the first moment, when the users start the communication. The users can select different codec for speech, but it is recommended use the G.723 [11] is predefined by NetMeeting application

1.2.2 Video

If the users like to have a video conference, the users must have a Webcam connected and working before starting the NetMeeting. When the user has established a communication, he can select the option to send video to the other user or users in the conference. The option is in menu item Tools / Video / Send. The users can choose also if they like to have video of the other participants in the conference.

It is possible to see also the own video that has been sent to the other/s in the conference in real time. The user has two windows, one with the video of the colleague and other one with the image of himself.

Figure 1-2 NetMeeting presents a image in a Web cam conference.

1.2.3 Desktop sharing

The NetMeeting allows to share desktop or other application that the users like. This service has all their power in a help desk service help a friend, or virtual meeting. The users can share a program or desktop to get help. To start the service the user has to stay inside of a conference (the connection must be established). Press the button named share program show in Figure 1-1. The Figure 1-3 shows the screen that was shown after clicking the share program button in the main screen. The user has to select a program and press the button

'Share'. When this occurs and Desktop sharing was selected, the user in the other side of the conference see the Desktop of his colleague as can be seen in Figure 1-4. If the user selects other application, his colleague can only see that application and all other windows are hidden for him. When the user share the entire desktop it is possible to see all the applications that are running at the moment (the task list shown in Figure 1-3). There are two types of Desktop sharing. The first is 'see only' mode. The other mode is that the user watching the desktop can perform actions or orders on the shared computers.

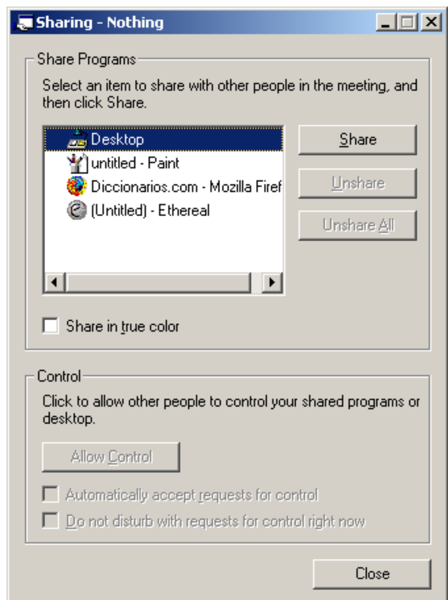

Figure 1-3 Sharing application in NetMeeting

The Figure 1-4 shows an example of the desktop sharing. It is possible to see a Desktop of a computer with windows 2000 and it is possible to see the NetMeeting application and 'My Computers' explorer window.

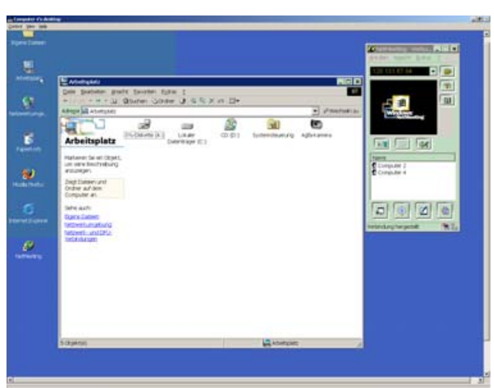

Figure 1-4 Desktop sharing.

1.2.4 Chat

The Chat service is based on the interchange of text messages of between users that have been connected to the conference. The Figure 1-1 shows two users, one of these users can start the chat application. When the user press the button, the application starts and show a picture presented in Figure 1-5.

The Figure 1-5 shows that the participant in the conference, can send message to all users or private message to one user (private chat).

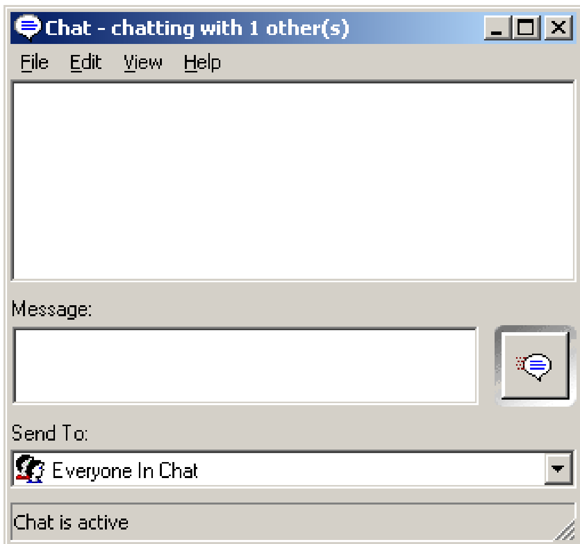

***Figure* 1-5** Chat Application

1.2.5 Whiteboard.

With the NetMeeting it is possible to share an MS Paint like application named WhiteBoard. This application has less functions that the original MS Paint, but is a complete program to draw different objects.

The user has to press the button named Whiteboard shown in Figure 1-1. When this occurs the program screen appears as shown in the Figure 1-6.

The figure 1-6 shows the Whiteboard application. It is possible to see a drawing in the Figure 1-6. The participants can paint at the same time within the picture.

Figure 1-6 Whiteboard application

1.2.6 Transfer a file

The figure 1-1 shows the button 'Transfer file' that the user has to press to share or send files.

When the user press the button, the application show a dialog box to select the file that like to share and the user that is possible to share it. This service can be private or broadcast service. The user has to select to destination user in the respective dialog box.

The Figure 1-7 shows one file that has been selected and sent to other user.

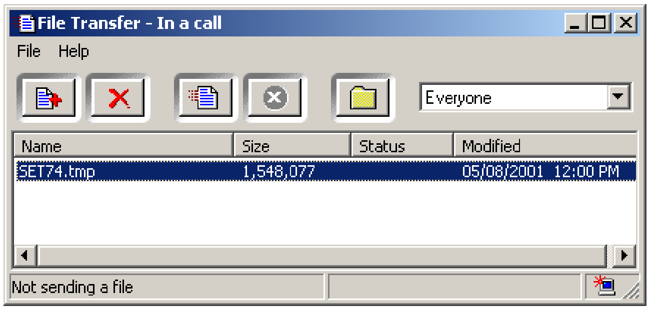

Figure 1-7 Transfer file.

1.3 Protocol Stack

The NetMeeting functions are based on several public standards of the International Telecommunication Union (ITU).

The Figure 1-8 shows how the standards work together to establish the different service, it is possible to see the different layers offering the service to the upper layer. All the protocols that use NetMeeting are standards and for this reason this software can work with other program of different companies for example Gnomemeeting (free program that works under linux).

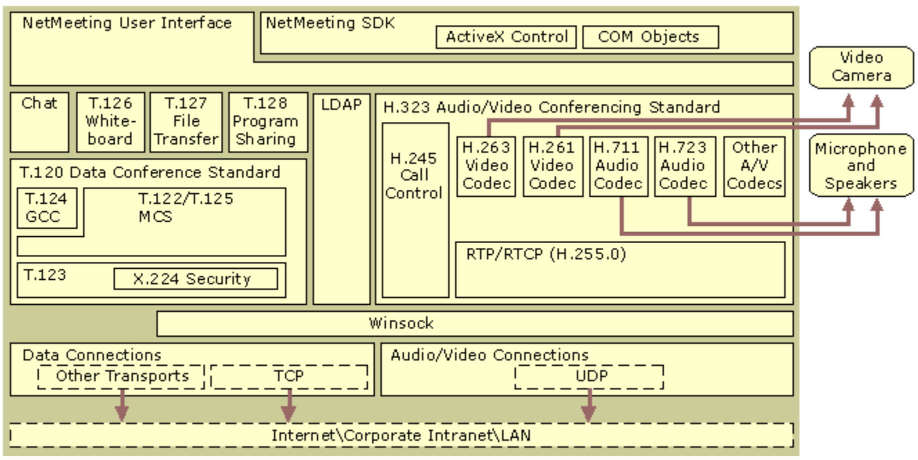

Figure 1-8 Protocols that are used by NetMeeting [1]

1.3.1 T.120

This protocol makes sure that the participants in the conference can receive the data without error. The protocol works over different protocols in the transport layer, between them TCP.

The protocol supports different kind of topologies if the conference is established with more than two participants.

The ITU has defined two types of protocols, networking protocols and application protocols. The networking protocol is T.120 and the applications are T.126, T.127 for example.

The networking standard has a control of flow data transported of NetMeeting communication.

The architecture of T.120 is based on a standard of the OSI. The Figure 1-9 shows the different layers of it and how this different layers interact between each other.

1.3.2 T.121

The function of T.121 is to get information about the protocols that the other peer support. It is possible that one peer understands the H.263 for video but the other peer does not understand. The function of the T.121 is to negotiate the capabilities of the end terminal.

- Registers the user in the conference.
- Applies the capabilities of the program to the other users.
- Negotiate the capabilities of the program.

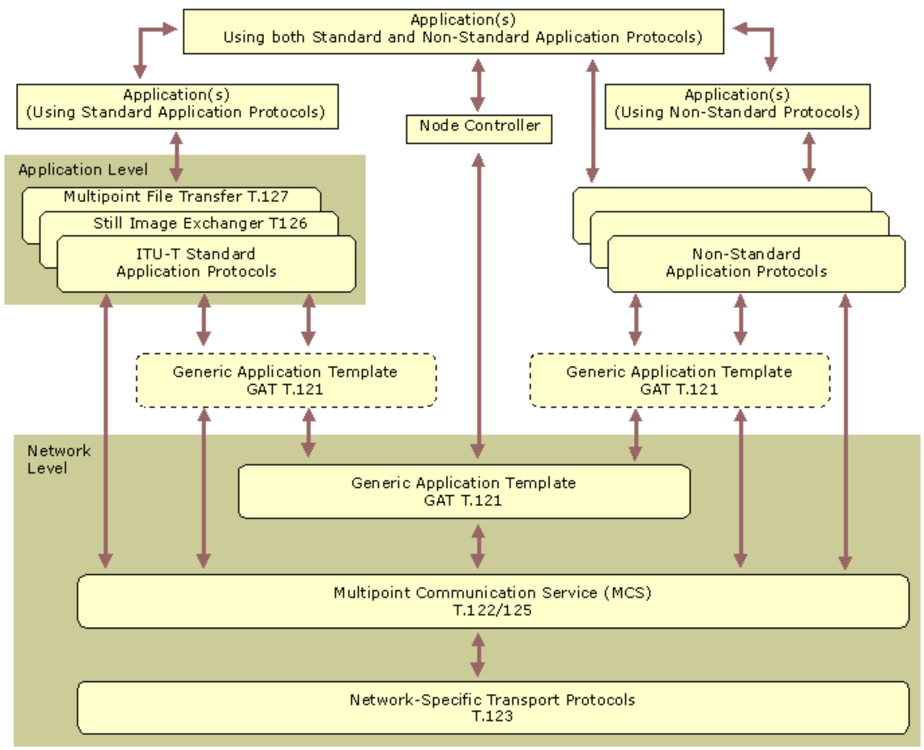

Figure 1-9 T.120 architecture
[3]

1.3.3 T.122

This protocol allows a conference with one or more participants. All the participants can send data to the other participants of course. This protocols work together with the protocol T.125, because the multipoint services are implemented in this protocol. The T.122 supports different topologies. The figure 1-10 shows the different topologies.

1.3.4 T.123

This protocol is responsible for the security of the communication and transporting sequence data and controlling the data flow over the networks. The functions include connect, disconnect, send and receive functions.

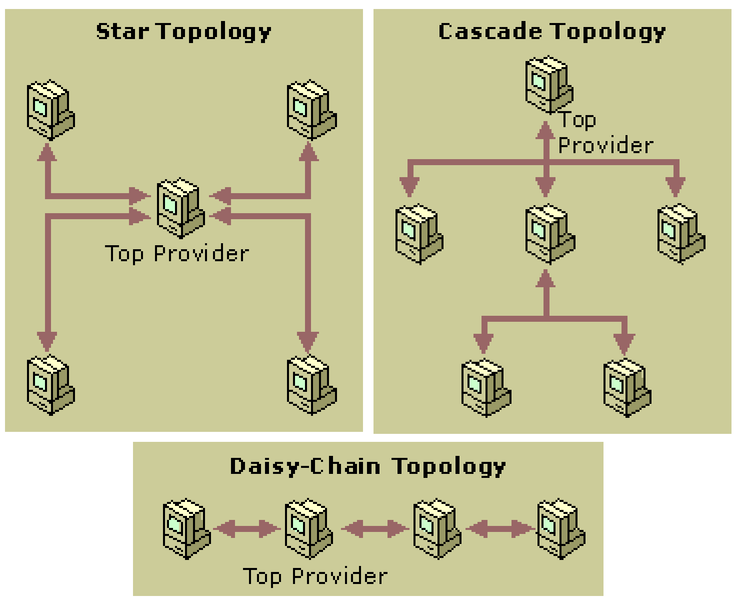

Figure 1-10 **Different topologies according to [2]**

1.3.5 T.124

This protocol provides a generic control conference protocol for initiating and admin multipoint conference.

Know the users that there are in the conference, and their applications. It has a control of the data and makes a monitoring of the data.

1.3.6 T.125

This standard says how the data is transported in the conference. This layer uses the service of layer T.122.

The layer defines the private and broadcast channels that transport the data, and ensures accurate and efficient communication among multiple users.

1.3.7 T.126

This protocol defines how the information of the whiteboard application is sent. For example is the information is compressed or uncompressed

1.3.8 T.127

This protocol allows to transmit one or more files between different user that has been selected while the conference has place.

1.3.9 T.128

This protocol was proposed by Microsoft and accepted by ITU as a standard named T.128. This protocol defines how a program can be shared to other computer. This protocol defines how many participants can share data, defining it in T.120. The number of user comes defined by the capabilities of the program.

Chapter 2. Scenario

The Figure 2-1 presents the scenario of the emulation. The scenario consists of 3 computers inside of a LAN. Two of them (clients) have NetMeeting installed and working. The other one has an ethereal application to sniff the traffic that has been generated by NetMeeting. The other computer has the application Dummynet installed. This program allows to emulate the bottleneck that is typical for the radio access network of mobile communication systems like universal mobile telecommunications system (UMTS). With Dummynet one can set the bandwidth, queue length, delay of the link and packet loss probability.
The scenario has been divided in two parts. The first part having a public IP is the computer with subnet number 128.131.0.84 The other part of the network has a private IP with subnet 192.168.0.50. In the middle there are a computer that make the translation of IP and has installed the dummynet

[7] application. All of this is transparent to the user.

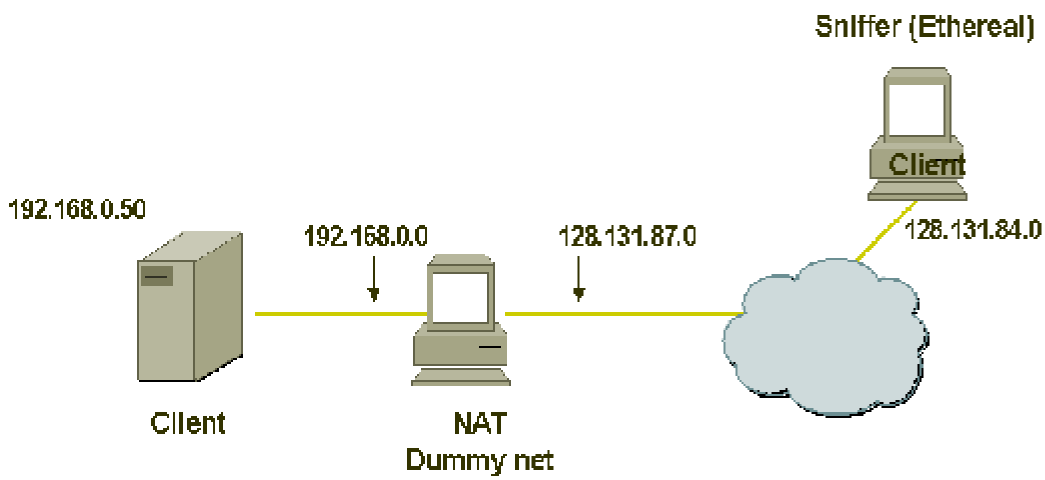

Figure 2-1 Experimental setup for delay and bandwidth limitations emulation

2.1 Packetization of H.263

The video call in NetMeeting is a real time one or two-way communication, for which especially the delay and delay jitter are important from the point of view of end-user quality.

The H.263 is transported as a payload of the RTP packets. This is a normal transport of real time communication inside of IP network, for example internet.

The type of packets generated by NetMeeting for the video traffic is described in standard
[4].

The data is transported using the different layer. RTP header/H.263 payload header / H.263 bitstream

```
No.     Time         Source              Destination          Protocol Info
719    17.692920 128.131.67.84    128.131.67.71       H.263    Payload type=ITU-
T H.263, SSRC=3396223223, Seq=51486, Time=34309260 MODE A

Frame 719 (1028 bytes on wire, 1028 bytes captured)
Ethernet II, Src: 00:02:1e:f2:4d:d1, Dst: 00:50:ba:a1:ad:4a
Internet Protocol, Src Addr: 128.131.67.84 (128.131.67.84), Dst Addr:
128.131.67.71 (128.131.67.71)
User Datagram Protocol, Src Port: 49606 (49606), Dst Port: 49606 (49606)
Real-Time Transport Protocol
10.. .... = Version: RFC 1889 Version (2)
..0. .... = Padding: False
...0 .... = Extension: False
.... 0000 = Contributing source identifiers count: 0
0... .... = Marker: False
.010 0010 = Payload type: ITU-T H.263 (34)
Sequence number: 51486
Timestamp: 34309260
Synchronization Source identifier: 3396223223
ITU-T Recommendation H.263 RTP Payload header (RFC2190)
F: False
p/b frame: False
Start bit position: 0
End bit position: 5
SRC format: CIF 352x288 (3)
Inter-coded frame: False
Motion vector: False
Syntax-based arithmetic coding: False
Advanced prediction option: False
Reserved field: 0
Differential quantization parameter: 0
Temporal Reference for B frames: 0
Temporal Reference for P frames: 233
H.263 stream: 000083A60E023FFFFF00008416C6D888...
```

Figure 2-2: Example of packet sent by NetMeeting and captured by Ethereal

The Figure 2-2 shows the structure of the H.263 packet that was captured by Ethereal and has the structure as specified in the H.263 standard.

The standard H.263 specifies different types of packetization. NetMeeting only uses one of them. The mode used is mode A with only I and P frames (no B frames).

In mode A, an H.263 payload header of four bytes is present before the actual compressed H.263 video bitstream. It allows for fragmentation at group of blocks (GOB) boundaries.

The Figures 2-3 and 2-4 show the characteristics of the video communication that NetMeeting generates using video. The Figure 2-3 shows the histogram of the packet sizes and the Figure 2-4 is a histogram of the delay between consecutive packets. The Packet sizes are less than 1500, the reason is that the application send a packet when it has ready one. The application does not wait to have a complet packet.

The bit rate that the channel needs for this communication was 368 kbit/s and the picture resolution was 352x288, called CIF.

NetMeeting allows to use also other type of video codec
[8]. It is possible to read in Resource Kit NetMeeting 3.1. But Microsoft recommends H.263.

We can write the probability of particular sizes of packets and inter packet delays as follows:

Packet size

$$PDF(X) = \frac{1}{\sqrt{2\pi 400^2}} \exp^{-\frac{(X-1400)^2}{(2*400)^2}} * G(X) \qquad \textbf{(2-1)}$$

$$G(X) = \begin{cases} 1 & 1 \le X \le 1400 \\ 0 & X > 1400 \end{cases}$$

Delay

$$P(X) = \begin{cases} 0.8 & (X = 0) \\ 0.2 & (0.1 < X < 0.9) \end{cases} \qquad \textbf{(2-2)}$$

Delay has a peak in 0 and then we can approximate the rest with the uniform distribution in the interval between the 0.1 and 0.9.

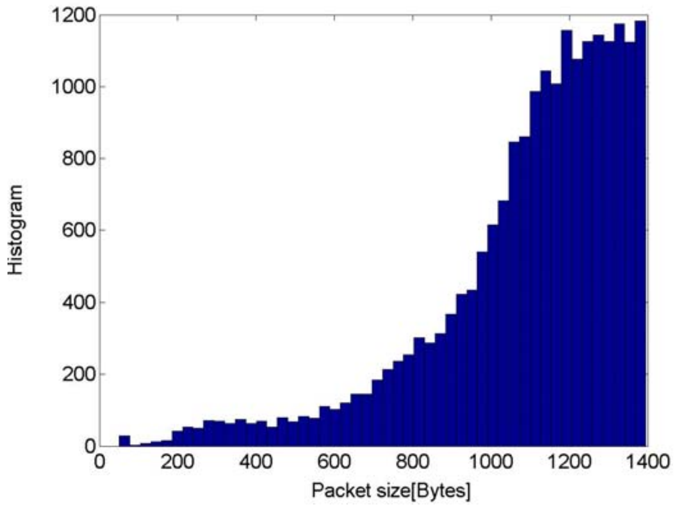

Figure 2-3 Packet size distribution for NetMeeting videoconference using H.263

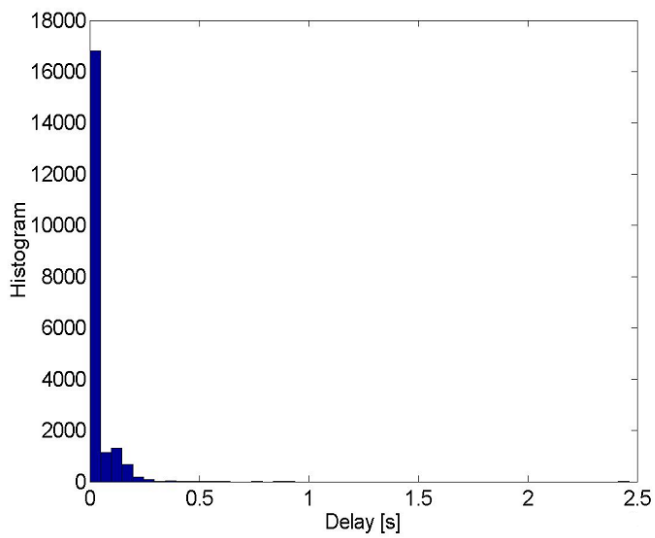

Figure 2-4 Inter packet time distribution for NetMeeting videoconference using H.263

2.2 Speech

The NetMeeting uses different standards for speech codecs. The documentation of NetMeeting

[8] says without any further explanation that the best codec of audio that NetMeeting can work with is Packet Voice Protocol (PVP) specified in ITU G.764.

Inside of the ITU G.764 there are two types of codec: PCM or ADPCM.

The convert the speech from analog to digital format one has to perform following three steps: sampling, quantification and codification.

ADPCM is a variant of PCM that uses variable quantization step size, reducing the necessary bandwidth.

For the speech traffic with PCM G.724 codec the packet size and inter packet delay distribution can be seen in the Figure 2-5 and 2-6. The packet size is always the same: 36 bytes of information. The delay between two packets has a maximum value of 30 ms. The delay of 30 ms is not perceptible for a human. NetMeeting uses UDP transport protocol to transmit the speech in real time. There is a bit rate of 28 kbit/s necessary in full duplex channel. The standard says that the maximum time stamp allowed in the packet is 200 ms. If the time of the packet is bigger, it will be discarded. In this case it is possible to see in the histogram that the delay is always less than this number. The size of the packet depends on the codec (PCM or ADPCM) that use NetMeeting and the sample rate.

Speech codec generates the packets with the same size and therefore the probability of this size is one:

$$P(X) \approx 1 \qquad X = 65 \tag{2-3}$$

The delay distribution we can approximate by following probabilities:

$$P(X) = \begin{cases} 0.65 & (X = 30) \\ 0.35 & (0 < X < 30) \end{cases} \tag{2-4}$$

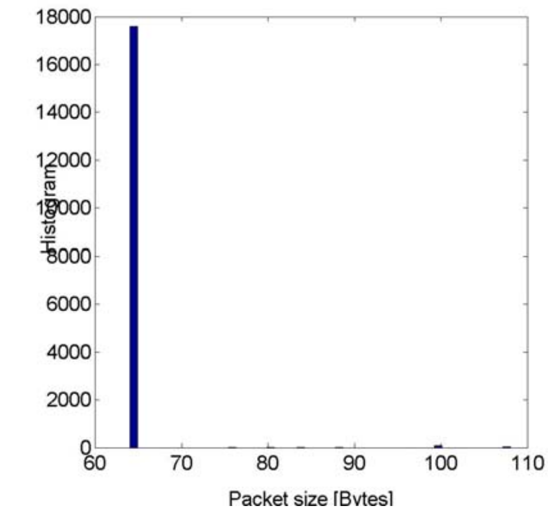

Figure 2-5 Packet size distribution for NetMeeting speech using PCM

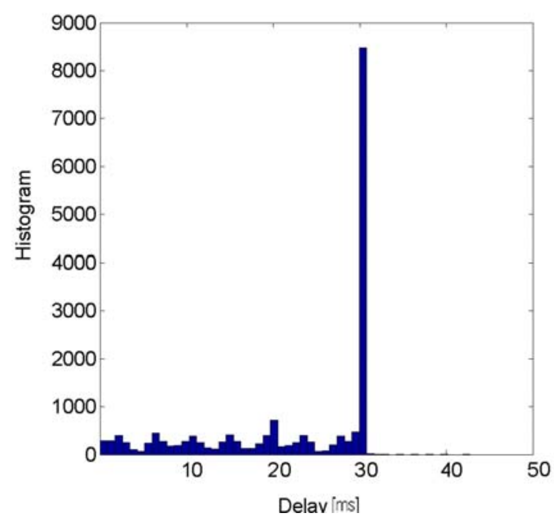

Figure 2-6 Interpacket delay distribution for NetMeeting speech using PCM

2.3 Desktop sharing

NetMeeting uses the T.128 standard of the ITU to share desktop or any application in the computer with one or more partners. The Figure 2-7 and 2-8 show only the communication between two computers. The Figure 2-7 it is possible to see the different sizes of the packets generated by this protocol. NetMeeting uses TCP protocol to transport the data. These protocols send ACK to confirm the correct receiving of the data – it is a reliable protocol.

The others are the info that sends NetMeeting to refresh the communication or different information to update the desktop or the application only when the user move the mouse or do some type of activity. If the user does not make any type of activity, NetMeeting does not send any information of refresh.

The average value of the bit rate that NetMeeting needs for this application is 64 kbit/s.

The protocol allows negotiation of the type of compression that the program likes to use. There is no mandatory compression protocol, but it is possible to use V.42 bis. In this protocol the data compression procedures for data circuit-terminating equipment (DCE) using error correction procedures are specified

The compression can be applied to the orders or to the bitmap. Both types of compression have a good ratio of compression.
The maximum compression of V.42bis is 4:1.

The probability density function of the packet sizes can be approximated by following function:

$$P(X) = \begin{cases} 0.5 & (X = 1500) \\ 0.2 & (X = 1200) \\ 0.3 & PDF(X) = \lambda e^{-\lambda x} \qquad \lambda < 500 \end{cases} \tag{2-5}$$

The probability of the delay is as follows

$$P(X) \approx 1 \qquad X = 0 < 0.030 \tag{2-6}$$

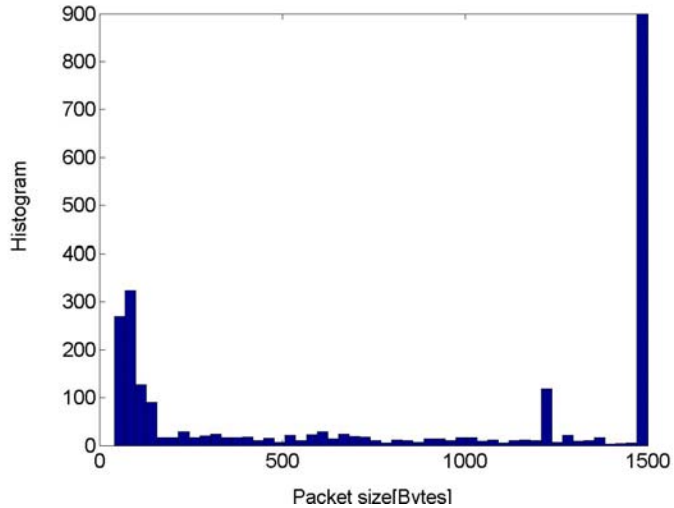

Figure 2-7 Packet size distribution for NetMeeting‘s Desktop Sharing

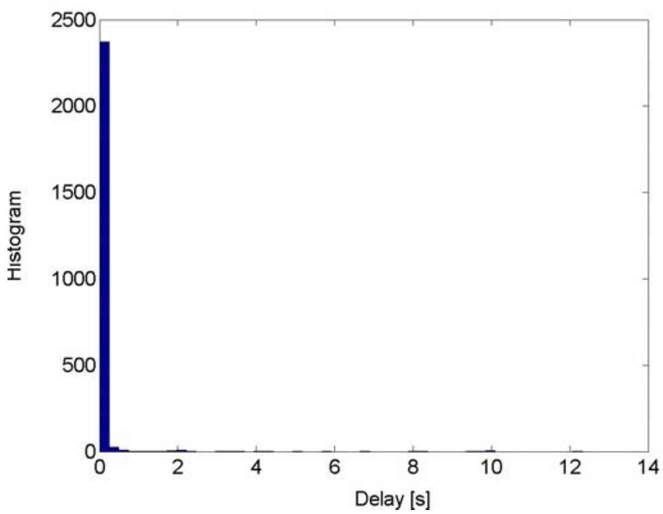

Figure 2-8 Inter packet delay distribution for NetMeeting’sDesktop sharing

2.4 Transfer File

An example of traffic generated by NetMeeting when using the transfer file between two computers.

```
No.     Time        Source               Destination          Protocol Info
  1812 57.360177   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1265017 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1813 57.360196   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1266477 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1814 57.360206   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [PSH, ACK] Seq=1267937 Ack=109 Win=65427 [CHECKSUM
INCORRECT] Len=1133
  1815 57.361509   128.131.67.73        128.131.67.84        TCP      1503 >
1576 [ACK] Seq=109 Ack=1267937 Win=65535 Len=0
  1816 57.391410   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1269070 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1817 57.391425   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1270530 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1818 57.391437   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [PSH, ACK] Seq=1271990 Ack=109 Win=65427 [CHECKSUM
INCORRECT] Len=1133
  1819 57.392447   128.131.67.73        128.131.67.84        TCP      1503 >
1576 [ACK] Seq=109 Ack=1270530 Win=65535 Len=0
  1820 57.392829   128.131.67.73        128.131.67.84        TCP      1503 >
1576 [ACK] Seq=109 Ack=1273123 Win=65535 Len=0
  1821 57.422669   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1273123 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1822 57.422685   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [ACK] Seq=1274583 Ack=109 Win=65427 [CHECKSUM INCORRECT]
Len=1460
  1823 57.422696   128.131.67.84        128.131.67.73        TCP      1576 >
1503 [PSH, ACK] Seq=1276043 Ack=109 Win=65427 [CHECKSUM
INCORRECT] Len=1133
  1824 57.423949   128.131.67.73        128.131.67.84        TCP      1503 >
1576 [ACK] Seq=109 Ack=1276043 Win=65535 Len=0
```

Table 1: File Transfer log captured by Etherreal

The kind of traffic that NetMeeting generates is a normal TCP communication. There are 4 types of packets. Three of them are info of data and one ACK packet of answer

The push flag in the TCP packet is used to force to send the data to the upper layer. With this the program tells that all the info is fine and forces all the buffered data to the receiving application. The graph shown in Figure 2-9 contains three tips. The two tips more on the right correspond to data sent by the computer that are transporting the user information. The other tip is negligible. This delay is higher than for the previously described applications. Video conferencing, speech and desktop sharing are real time applications and therefore the user requirements to the delay are higher. With file transfer the delay is not critical, one only has to ensure, that the delay caused by network will not lead to the TCP time out and resulting connection failure.

In the Figure 2-9 it is possible to see a three peaks in the distribution of the packet sizes with a probability of:

$$P(X) = \begin{cases} 0.65 & (X = 1500) \\ 0.35 & (X = 1200) \end{cases} \qquad \textbf{(2-7)}$$

The value of stay in 0 is 1 for the distribution of inter packet delay.

$$P(X) \approx 1 \qquad 0 < X < 0.020 \qquad \textbf{(2-8)}$$

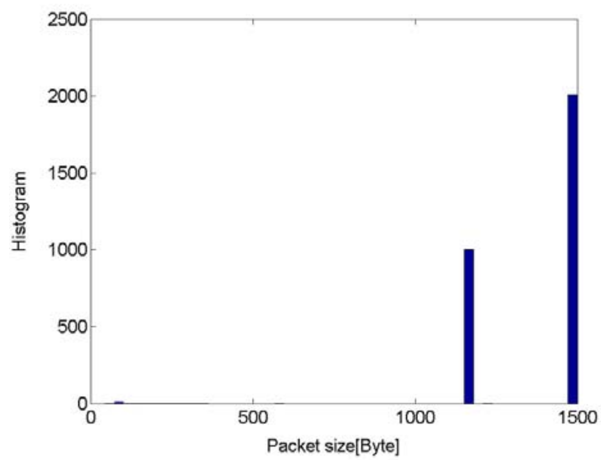

Figure 2-9 Packet size distribution for NetMeeting for transfer file

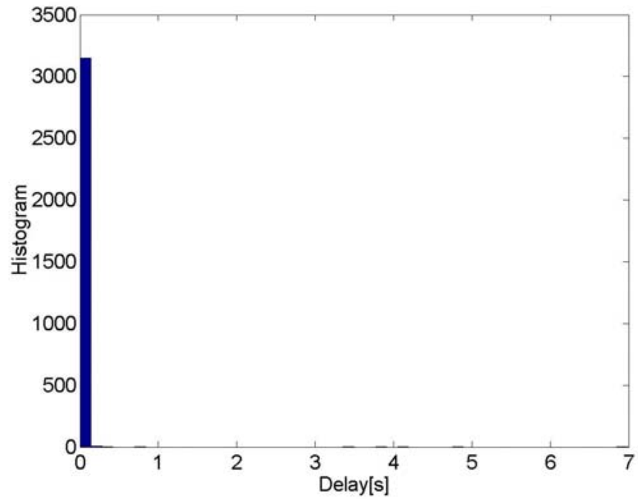

Figure 2-10 Inter packet delay distribution for NetMeeting for transfer file**.**

2.5 Whiteboard.

In this case there are two computers one of them opens the whiteboard and both users can paint in it. The traffic that generates NetMeeting is based on T.126 standard of the ITU. The standard specifics how the program sends the information and the updates to the other computers. The updates of the information is only when one of the users do some activity. If the user do not send any kind of information no updates are sent. The whiteboard needs in average bitrate of 7 kbit/s.

The Figure 2-11 shows the packet size histogram, that contains three peaks and can be approximated by following function:

$$. \ P(X) = \begin{cases} 0.35 & (X = 50) \\ 0.55 & (X = 100) \\ 0.10 & (X = 1150) \end{cases} \qquad \textbf{(}2\text{-}9\textbf{)}$$

The Figure 2-12 shows one discrete function present in the next formula

$$P(X) = \begin{cases} 0.8 & (X = 0) \\ 0.2 & (0.1 < X < 0.9) \end{cases} \qquad \textbf{(}2\text{-}10\textbf{)}$$

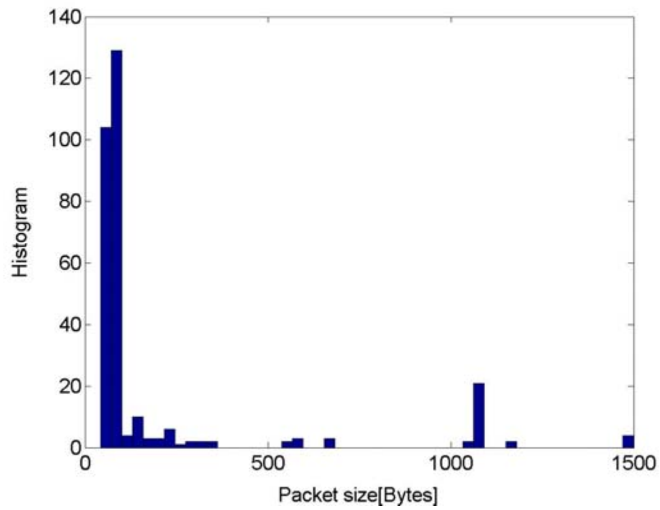

Figure 2-11 Packet size distribution for NetMeeting for whiteboard

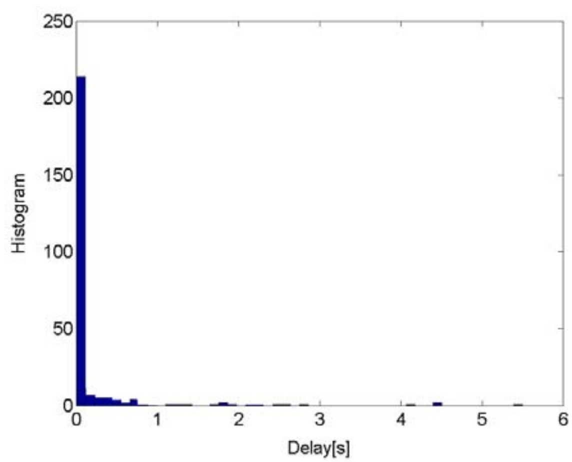

Figure 2-12 Inter packet delay distribution for NetMeeting for whiteboard**.**

2.6 Chat

The chat specification is based on ITU T.120 specification

[6]. The text that has been written with NetMeeting is sent by TCP packet. This TCP packet has a maximum length of 1500 bytes. If the length of the user message is bigger than the maximum length the information is fragmented in two packet. The value when the packet is fragmented and compressed is 2500 characters of info. The delay of this communication is not so important. The reason is that the communication is not in “real time”. The delay acceptable in this communication is bigger, because the user can be answering other communication at the same time and the other user has to wait. The user introduces the delay in the network. The value of the delay like is possible to see is the biggest in all the protocols than has been seen. It has been found that NetMeeting use H.425 to transmit the chat message, but this protocol is used as a transfer protocol for multimedia communication. The standard of ITU to transmit text chat is T.140.

There are no Distributions of the Chat. The reason is that the values that has been captured of the network and generated by NetMeeting is two much random. For each type of communication the values of the packet is different. The reason is that the packet always by bigger than 28 bytes of data that NetMeeting transmit in each packet. The information of the data is the message is a private or public and the user how write it. NetMeeting always send a packet when the user press enter, in this moment a TCP is generated to the other computer, with the size of the phrase that the user write. The test was made over normal conversation of 5 minutes, trying to simulte it. And the values of the packet and the delay by too much different according to the last messaurament.

Chapter 3. MOS measurements.

3.1 Programs

To complete this projet I used three programs: The Ethereal used to capture the packets, Dummynet that allows emulating the bottleneck and NAT that allows Dummynet handling as a router between the client and the LAN.

3.1.1 Network Address Translator (NAT).

The function of a NAT application is to translate the address of the internet to an intranet or a private IP network. Each IP address in internet must be unique. This is a logical requirement is the same that if a telephone company has two phone numbers with the same numbers. The network does not know with what number has to establish the communication.

The necessity of this program comes given that there are more computers in internet than public address available. The function of the NAT is just this. Simulating that there are more IP address that there really are in internet. Exactly the function of the program is so that with one Public IP it is possible to connect more computers that have a private network. To make this it is necessary that one computer will be handling as the bridge between the public and private network. This is the function of the NAT.

The reason that NAT application is used in my project is that dummynet application intercepts the packet of NAT and uses the packet to simulate the state of the network, delay, and bandwidth.

The dummy net application is explained later.

The Figure 3-1 shows a possible NAT configuration. There are three computer inside of the private network. There are one NAT that make the function of NAT (to translate the address of the communication) and the other side is internet for example.

The computer 192.168.0.51 establishes a communication between it and the computer named www.tuwien.ac.at or 128.131.102.130. The communication start from the computer 192.168.0.51 to NAT computers, and the nat computers to www.tuwien.ac.at . All this process is transparent to the user.

There are different solutions to implements a NAT configuration. This is only an idea to the necessity of the NAT.

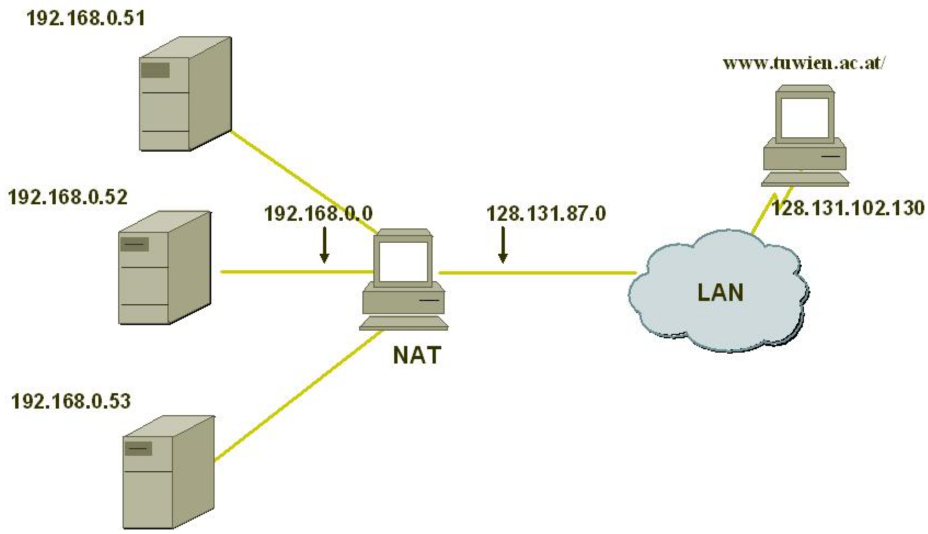

Figure 3-1 Scenario of a possible NAT configuration**.**

3.1.2 Ethereal

Ethereal
[5] is a network analyzer based on a software application. The program works in a windows or Linux environments.

When the application is started it is possible to see 5 different parts of ethereal program in the screen.

Principal Menu. Show all the options that Ethereal has. In the majority of the case It will be use to analyse all the traffic.

Zone of results Show the packet sniffed of the network.

Selected Frame Show the content of the frame selected.

Frame selected without formatted Show the frame without format, as it is captured.

Load of filters It is possible to applied the filters directly in the info.

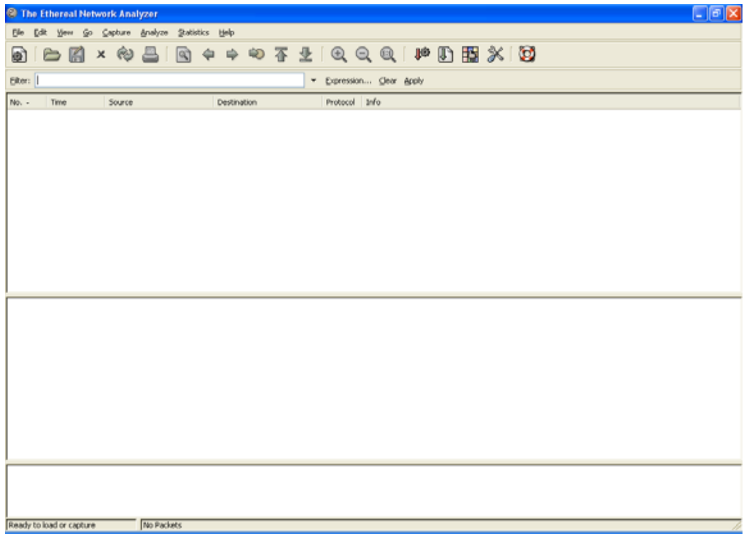

Figure 3-2 Ethereal application

The menu item ‘capture’ has two sub options. One of these options is start. When this option is selected, the next screen is shown. It is possible to choose different options in the traffic. All the possible are available in the figure.

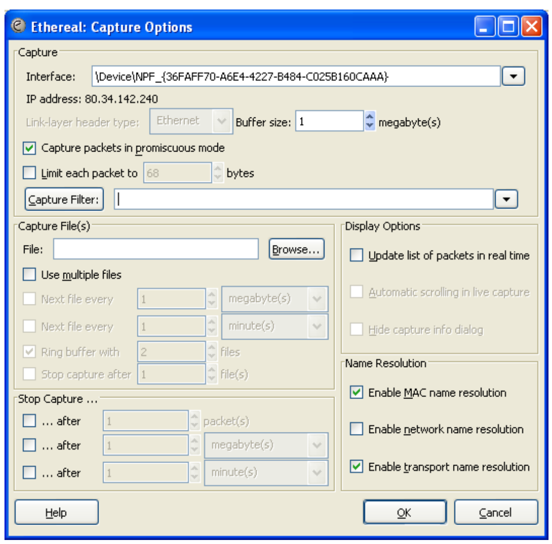

Figure 3-3 Capture Options in Ethereal.

When all the options are selected and the OK button is pressed the program shows the next screen as shown in Figure 3-4. The figure shows the numbers of packets captured in the communication. It is possible to see that the most common or important feature implemented is to see how much packets are captured.

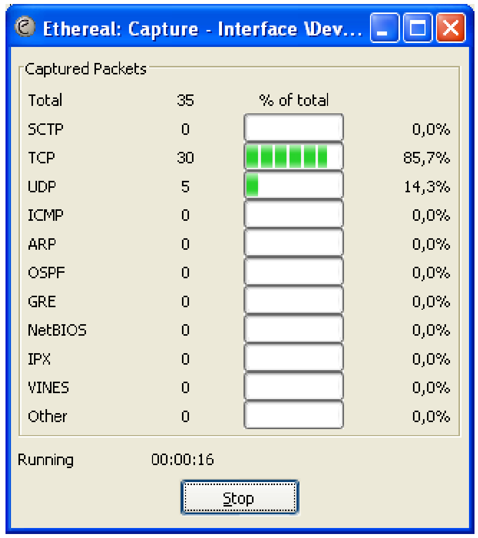

Figure 3-4 Capture traffic of network.

When the 'stop' button is selected the program shows in the screen the packet that has been captured.

The capture shows the traffic and it is possible to see the different parts of the screen that it commented before.

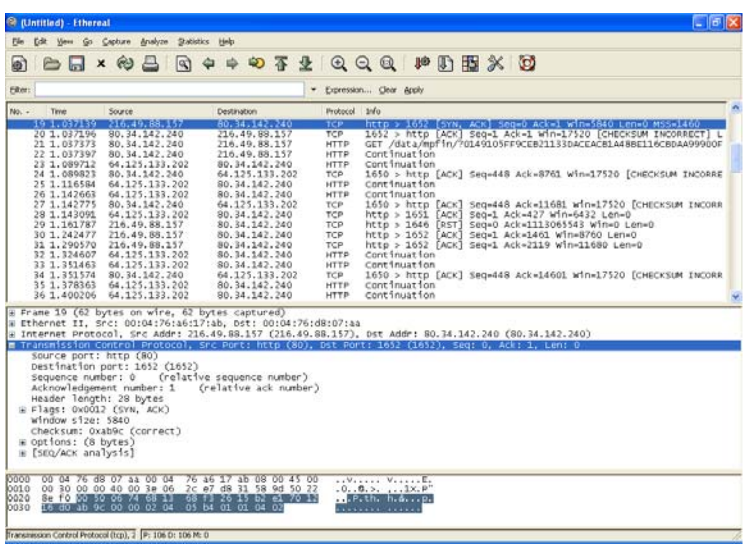

Figure 3-5 Ethereal screen with information of capture.

The filters are an other important function that ethereal application have to analyse the traffic. The filters allow to separate the traffic of the communication, differentiate the different protocol capture, or select the info for the size, source, destination, a flag of the protocol, and other options.

3.1.3 Dummynet

Dummynet is a Berkeley Software Distribution (BSD) program that allows to testing different characteristics of networks.

Dummynet works intercepting packets from the stack and passing them form pipes and queue to simulate the effects of the networks applied to ipfw command.

Each pipe and queue can be configured independent form the others. The protocols, ports, ranges, interfaces, delay, can be different for each parameters. The values of each pipe can be modify dynamically.

The most important characteristics used are: Delay, bandwidth, and queue length, but the program has other among then packet losses, mutipath effect.

The syntax of the command line are:

ipfw add pipe Number of pipe / protocol / from / to
ipfw pipe Number of pipe config bw XXX queue XXX delay XXX

example

ipfw add pipe 100 icmp from any to any
ipfw pipe 1 config bw 50 kbits queue 10 delay 10ms

The type of protocol that can modify are ICMP/UDP/TCP

To configure the scenario It's necessary the next sequence of command

ipfw add pipe 100 icmp from any to any
ipfw pipe 100 config delay 0ms queue 10

ipfw add pipe 200 ip from any to any
ipfw pipe 200 config delay 0ms queue 10

ipfw add pipe 300 tcp from any to any
ipfw pipe 300 config delay 0ms queue 10

ipfw add pipe 400 udp from any to any
ipfw pipe 400 config delay 0ms queue 10

ipfw pipe 200 config queue 20 delay 20 ms bw 64 kbps

This is the part of the scenario that is fix. Then it had to change the parameters of the communication between two computers. This is necessary to evaluate the subjective quality of the service received by the user.

3.1.4 Extract.exe

It is needed to extract the info that is sniffed with ethereal. To make easy this work has been made a program that extract the info of the size of each packet and the time that has been printed in an ASCII file by ethereal. The size of the files that ethereal generates is about 100 Mbytes in the case of the video info for example with more of thousands of packet.

The ethereal application print a file like:

```
No.     Time        Source                Destination
Protocol Info
     25 2.312427    128.131.67.84         192.168.67.120        TCP
1923 > 2331 [PSH, ACK] Seq=5 Ack=5 Win=65531 [CHECKSUM INCORRECT]
Len=529

Frame 25 (583 bytes on wire, 583 bytes captured)
    Arrival Time: Nov 19, 2004 11:42:33.260032000
    Time delta from previous packet: 0.147861000 seconds
    Time since reference or first frame: 2.312427000 seconds
    Frame Number: 25
    Packet Length: 583 bytes
    Capture Length: 583 bytes
Ethernet II, Src: 00:02:1e:f2:4d:d1, Dst: 00:50:da:43:1c:97
    Destination: 00:50:da:43:1c:97 (128.131.67.235)
    Source: 00:02:1e:f2:4d:d1 (128.131.67.84)
    Type: IP (0x0800)
Internet Protocol, Src Addr: 128.131.67.84 (128.131.67.84), Dst Addr:
192.168.67.120 (192.168.67.120)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN:
0x00)
        0000 00.. = Differentiated Services Codepoint: Default (0x00)
        .... ..0. = ECN-Capable Transport (ECT): 0
        .... ...0 = ECN-CE: 0
    Total Length: 569
```

```
    Identification: 0x9564 (38244)
    Flags: 0x04 (Don't Fragment)
        0... = Reserved bit: Not set
        .1.. = Don't fragment: Set
        ..0. = More fragments: Not set
    Fragment offset: 0
    Time to live: 128
    Protocol: TCP (0x06)
    Header checksum: 0x0000 (incorrect, should be 0x9b62)
    Source: 128.131.67.84 (128.131.67.84)
    Destination: 192.168.67.120 (192.168.67.120)
Transmission Control Protocol, Src Port: 1923 (1923), Dst Port: 2331
(2331), Seq: 5, Ack: 5, Len: 529
    Source port: 1923 (1923)
    Destination port: 2331 (2331)
    Sequence number: 5    (relative sequence number)
    Next sequence number: 534    (relative sequence number)
    Acknowledgement number: 5    (relative ack number)
    Header length: 20 bytes
    Flags: 0x0018 (PSH, ACK)
        0... .... = Congestion Window Reduced (CWR): Not set
        .0.. .... = ECN-Echo: Not set
        ..0. .... = Urgent: Not set
        ...1 .... = Acknowledgment: Set
        .... 1... = Push: Set
        .... .0.. = Reset: Not set
        .... ..0. = Syn: Not set
        .... ...0 = Fin: Not set
    Window size: 65531
    Checksum: 0xca23 (incorrect, should be 0x0f86)
    SEQ/ACK analysis
        This is an ACK to the segment in frame: 24
        The RTT to ACK the segment was: 0.000033000 seconds
DData (529 bytes)
```

Figure 3-6 Ethereal capture

The function of the program is extract and process the info. To make this, the program only need to introduce a name of the file that contains the data or information required to process, and the out of the program are generated two file named, Salida2.txt and salida3.txt. Salida2.txt has the size of the packet in the column 2 and salida3.txt has the different time between two packets of the communication.

The program is console line program written in C#.

extract entrada.txt

Where entrada.txt is the file that contained the info sniffed by ethereal.

The source code of the program is attached in the annex part of this document.

3.2 Measurements description

The measurements consist in trying the service that NetMeeting offers applying different delays and bandwidths and asking the test persons how they liked it. Test persons are asked to evaluate the quality of communication on a 10-grade scale. This is one of the possibilities recommended in [ITU-T p.910] for testing the quality of one way video. By averaging over the evaluations of all test persons we obtain mean opinion score (MOS) values.

The tested services are following

- Desktop sharing
- Chat
- Whiteboard
- Transfer File
- Speech
- Video conference

3.2.1 Desktop sharing

The first service that is proved is the desktop sharing. The idea is that in all steps every user performs the same process.

The NetMeeting interface can be seen in Figure 1-1. When the user press the first button on the left side a sharing window appears. In this window user can choose between the desktop and application sharing. If desktop is shared, all connected persons can see everything that the user is doing. If the application is chosen – the other connected users will only see that particular application. After choosing the sharing type and clicking ‘Share’ button, all connected users can see the desktop or the application of the user on their screens.

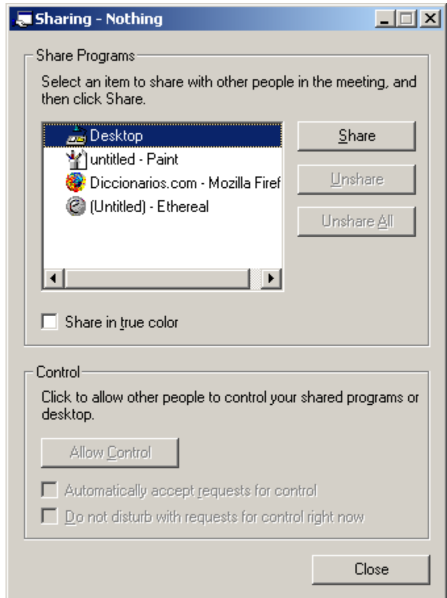

Figure 3-7 Sharing application in NetMeeting

In the tests we performed with the users, the first step is open a "MY COMPUTER" explorer window and select local Disk C: as can be seen in the Figure 3-8.

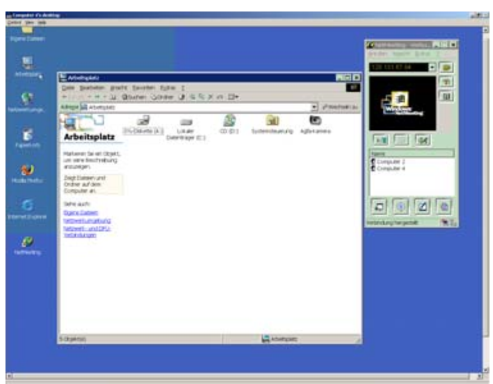

Figure 3-8 Desktop sharing.

Further they are asked to select the folder Winnt. Inside this folder there are a lot of files, and there are a lot of data transmissions. It's perfect to see the worst delay.

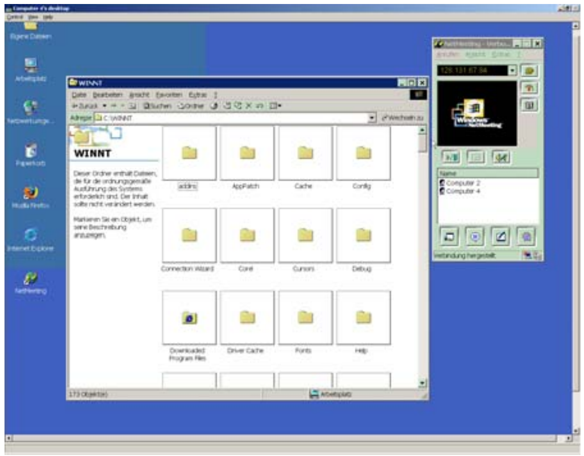

Figure 3-9 Desktop sharing.

3.2.2 Chat

In the chat session we ask the users to always write the same dialog. This is important for the delay and bandwidth perception that could be different depending on the content.

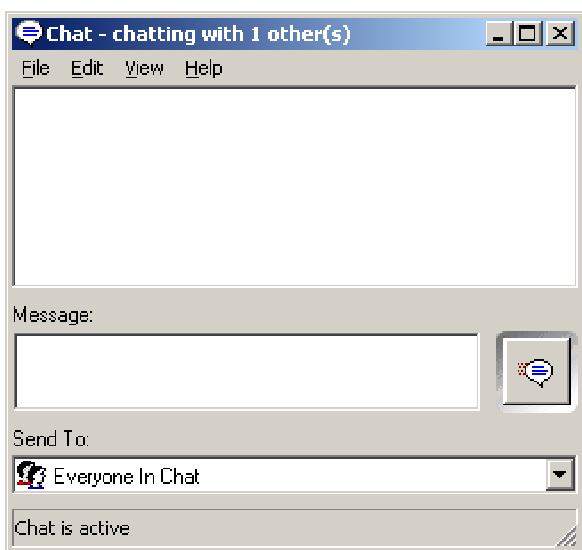

***Figure* 3-10** Chat Application

The texts that the user writes is a part of Othello act II scene III:

OTHELLO: Good Michael, look you to the guard to-night:
Let's teach ourselves that honourable stop,
Not to out-sport discretion.

CASSIO: Iago hath direction what to do;
But, notwithstanding, with my personal eye
Will I look to't.

OTHELLO: Iago is most honest.
Michael, good night: to-morrow with your earliest
Let me have speech with you.—Come, my dear love,— [To Desdemona]
The purchase made, the fruits are to ensue;
That profit's yet to come 'tween me and you.—
Good-night.

CASSIO: Welcome, Iago; we must to the watch.

IAGO: Not this hour, lieutenant; 'tis not yet ten o' the clock.
Our general cast us thus early for the love of his Desdemona; who
let us not therefore blame: he hath not yet made wanton the night
with her; and she is sport for Jove.

CASSIO: She's a most exquisite lady.

IAGO: And, I'll warrant her, full of game.

CASSIO: Indeed, she is a most fresh and delicate creature.

IAGO: What an eye she has! methinks it sounds a parley to provocation.

CASSIO: An inviting eye; and yet methinks right modest.

IAGO: And when she speaks, is it not an alarm to love?

CASSIO: She is, indeed, perfection.

IAGO: Well, happiness to their sheets! Come, lieutenant, I have a
stoup of wine; and here without are a brace of Cyprus gallants
that would fain have a measure to the health of black Othello.

3.2.3 Whiteboard.

The user presses the Whiteboard button, and appears a new window with a whiteboard.

This is the content that the user has to draw in the whiteboard.

Figure 3-11 Paint application

3.2.4 Transfer a file

The file that the user sends is set74.tmp. The size of the file is 1.548 Mbytes.

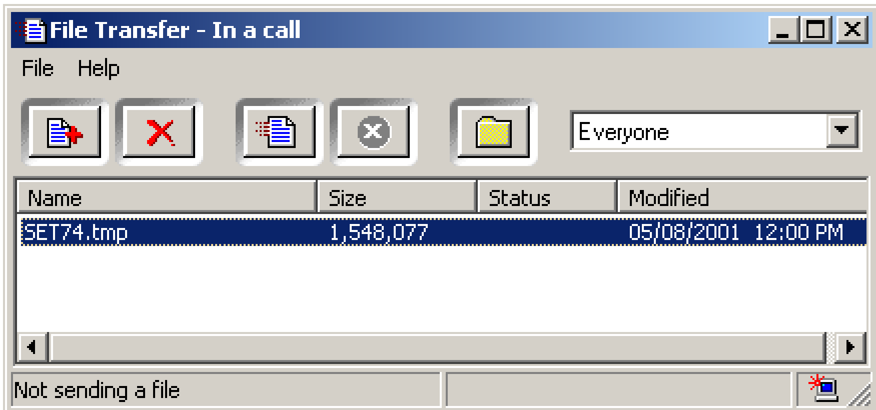

Figure 3-12 Transfer file.

3.3 NetMeeting: Subjective quality evaluation

The next part of the project is to measure the perception of the user for each type of traffic. In the delay of the communication is complete subjective for each user. The next tables are found the average value of the text to different users.

The scale goes from 0 to 10 being 0 the worst and 10 the best value.

For regression I used the CurveExpert [15] program.

Table 3-1 Average values of perceptual quality of speech

64 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
MOS	5,67	5,67	5,67	5,67	2,00	1,33	1,00	1,00	0,67	0,67
128 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
MOS	7,00	7,00	3,67	3,67	2,33	2,33	2,00	2,00	2,00	2,00
384 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
MOS	8,00	8,00	7,33	6,67	6,00	5,67	5,33	5,33	3,33	3,00

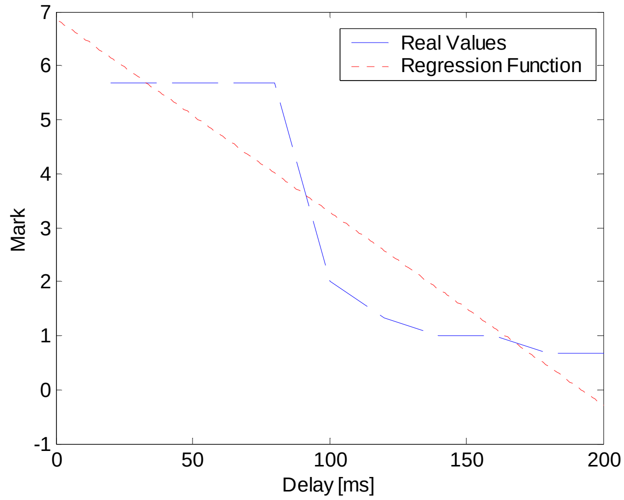

Figure 3-13 Show the subjective quality versus a regression function for Speech of 64 kbps.

The next equation shows the function of a linear regression that has been found to the speech of 64 kbps. The type of equation is linear Fit.

$$f(x) = \mathrm{a} + \mathrm{bx} \qquad \textbf{(}3\text{-}1\textbf{)}$$

The values of the Coefficient Data are:
a = 6.8693333
b = -0.035766667

Standard Error: 1.0550269
Correlation Coefficient: 0.9087403

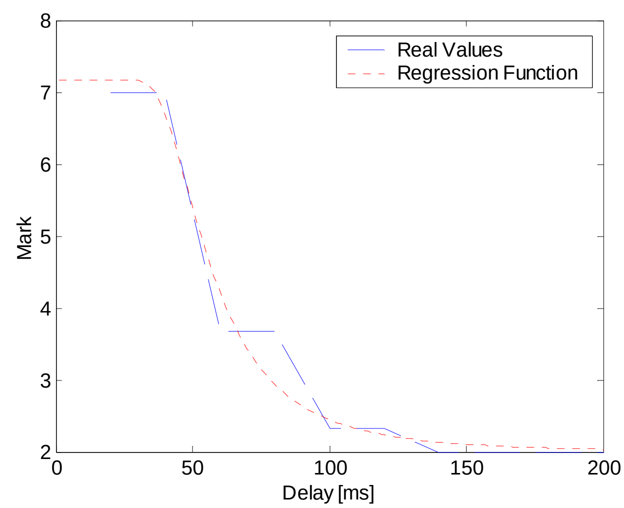

Figure 3-14 Subjective quality versus a regression function for Speech of 128 kbps.

The next equation shows the function of regression that has been found to the speech of 128 kbps. The function named Weibull Model

$$f(x) = a - b * \exp^{-c*x^{d}} \qquad \textbf{(3-2)}$$

Where the value of the coefficient data are:

a = 7.1650419
b = 5.1710933
c = 1219159.3
d = -3.5627694

Standard Error: 0.4102818
Correlation Coefficient: 0.9859495

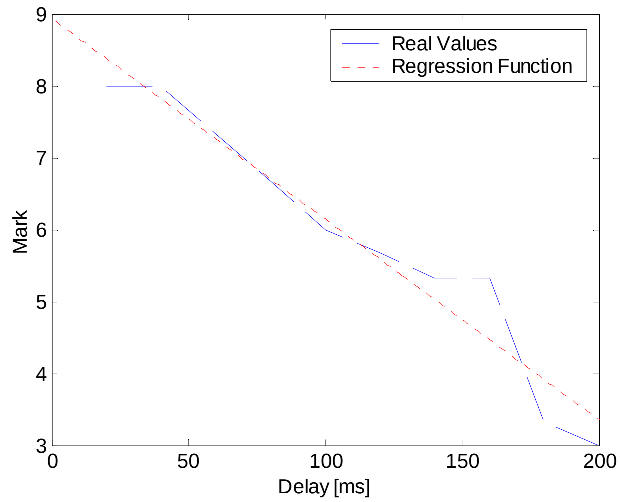

Figure 3-15 Show the subjective quality versus a regression function for Speech of 384 kbps.

The next equation shows the function of regression that has been found to the speech of 384 kbps. The type of equation is Linear fit.

$$f(x) = a + bx \qquad \textbf{(3-3)}$$

Where the value of the coefficient data are:

a = 8.934
b = -0.027890909

Standard Error: 0.4334523
Correlation Coefficient: 0.9719505

It is possible to extract of this information that with low delay the conversation can be possible in all the bandwidth measurable, but when the delay is bigger it fall down very fast. It curious according to the MOS that the graph of 64 kbps (Figure 3-13) has a better response than the graph of 128 kbps (Figure 3-14). It important to remember that is subjective to the user and the state of the network at this moment. The best case is logical is the 384 kbps that with a delay of 165 ms has a mark up of 5. It is logical that at more velocity the delay is less perceptible.

Table 3-2 Average values of perceptual quality of video

64 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
128 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	1,33	1,33	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
384 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	9,00	9,00	8,33	8,00	7,67	7,67	7,67	6,67	6,67	6,67

The is no Figure for the value of 64 kbps because it is zero for all delays which means that this bandwidth is not sufficient for transmission of the two way video service.

The next equation shows the function of regression that has been found to the Video with a bandwidth of 64 kbps is constant

$$f(X)=0 \quad \text{for all } X. \qquad \textbf{(}3\text{-}4\textbf{)}$$

In this case the bandwidth of 64kbps was not sufficient to transport the two-way video stream with CIF resolution.

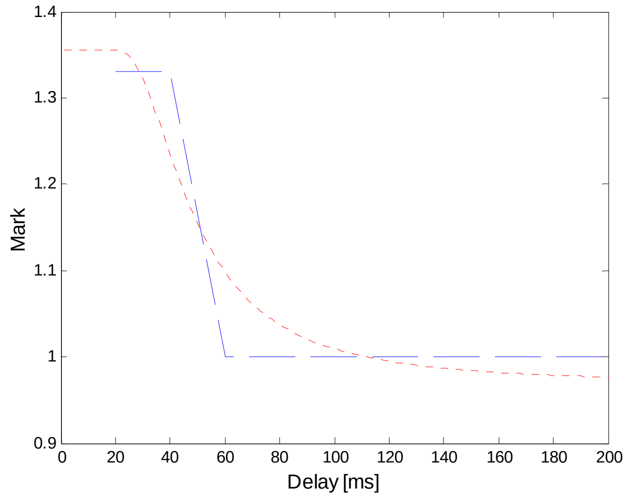

Figure 3-16 Show the subjective quality versus a regression function for Video of 128 kbps.

The next equation shows the function of regression that has been found to the video with a bandwidth of 128 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^{d}} \qquad \textbf{(3-5)}$$

Where the value of the coefficient data are:

a = 1.3557128
b = 0.38602714
c = 16432.624
d = -2.5907028

Standard Error: 0.0611336
Correlation Coefficient: 0.9334369

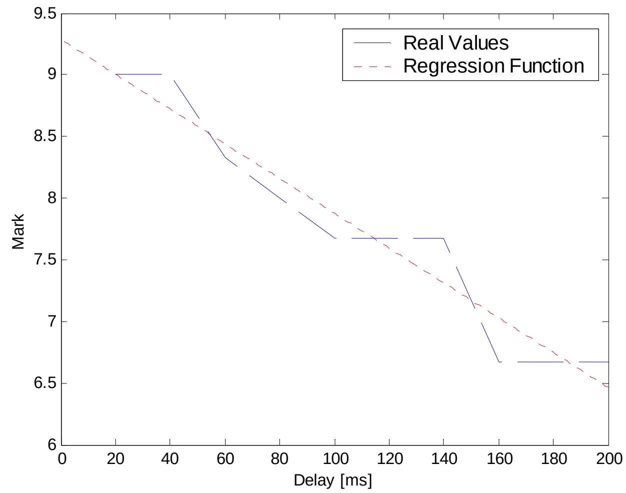

Figure 3-17 Subjective quality versus a regression function for video of 384 kbps.

The next equation shows the function of regression of the video with bandwidth 384 kbps. The name of the function is linear fit.

$$f(x) = a + bx \tag{3-6}$$

Where the value of the coefficient data are:

a = 9.2873333
b = -0.014112121

Standard Error: 0.2418768
Correlation Coefficient: 0.9661875

The MOS measurements in this case has a important values, for the users is not an acceptable service for a bandwidth under 384 kbps and in the case of 64 kbps the mark always was 0 in the all the person that make the measurements. In the case of 384 kbps it possible to have acceptable delay of 150 ms approximately. It is 5.

Table 3-3 Average values of perceptual quality of Whiteboard.

64 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	8,00	8,00	8,00	8,00	8,00	8,00	7,67	7,33	6,67	6,67
128 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	8,67	8,67	8,67	8,67	8,67	8,67	8,67	7,67	7,67	7,67
384 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	9,33	9,33	9,33	9,33	9,33	9,33	9,33	9,00	8,33	8,33

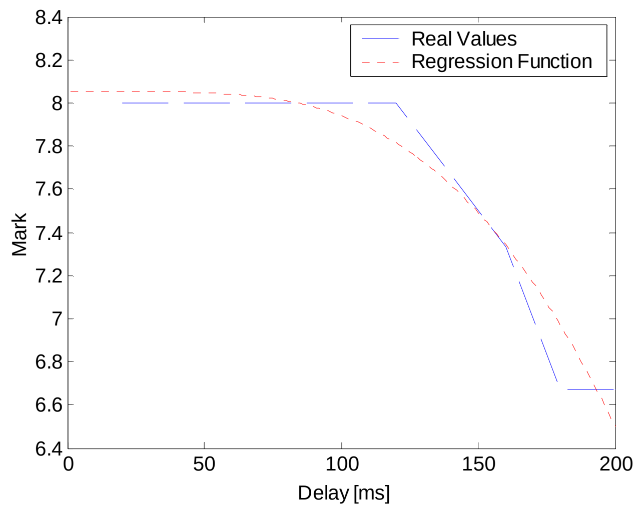

Figure 3-18 Show the subjective quality versus a regression function for Whiteboard of 64 kbps.

The next equation shows the function of regression that has been found to the Whiteboard with a bandwidth of 64 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^d} \qquad \textbf{(3-7)}$$

Where the value of the coefficient data are:

a = 8.0507547
b = 18480.818
c = 62.603261

d = -0.35812598

The error of correlation are:

Standard Error: 0.1663520
Correlation Coefficient: 0.9694096

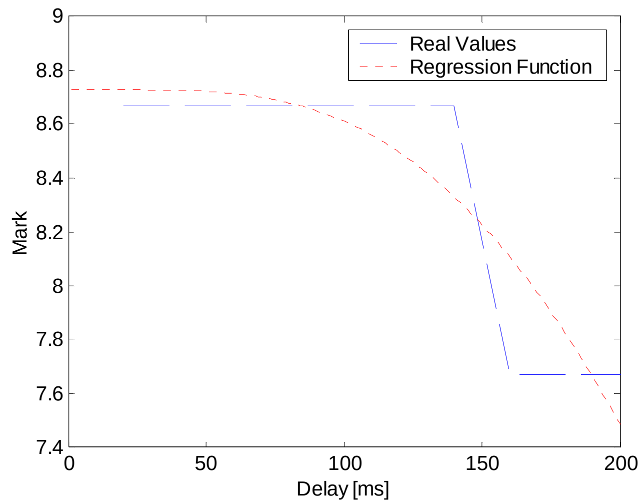

Figure 3-19 Show the subjective quality versus a regression function for Whiteboard of 128 kbps.

The next equation shows the function of regression that has been found to the Whiteboard with a bandwidth of 128 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^d} \qquad \textbf{(3-8)}$$

Where the value of the coefficient data are:

a = 8.7272426
b = 3968.547
c = 57.434758
d = -0.37036902

Standard Error: 0.2607649
Correlation Coefficient: 0.8976185

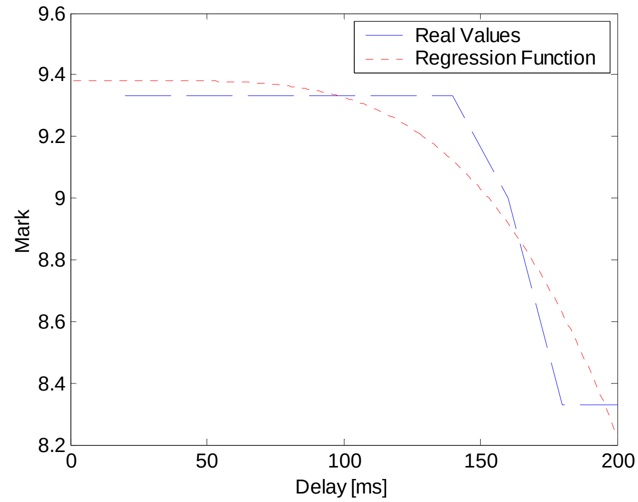

Figure 3-20 Show the subjective quality versus a regression function for Whiteboard of 384 kbps

The next equation shows the function of regression that has been found to the Whiteboard with a bandwidth of 384 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^d} \quad \textbf{(3-9)}$$

Where the value of the coefficient data are:
a = 9.3807489
b = 274173.22
c = 67.008415
d = -0.31872298

Standard Error: 0.1646795
Correlation Coefficient: 0.9466230

Acording to the MOS all the bandwidth are available to transmit this service. The different bandwidth that has been analyse it possible to extract that with 64 kbps is sufficient to extract the value, the delay is the most important to consider.

Table 3-4 Average values of perceptual quality of Transfer file.

64 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	1,00	1,00	1,00	1,00	0,67	0,67	0,33	0,33	0,33	0,33
128 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	3,00	3,00	2,33	2,33	2,33	2,33	2,33	1,67	1,67	1,67
384 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	6,67	6,33	6,00	5,67	5,67	5,00	4,67	4,33	4,00	4,00

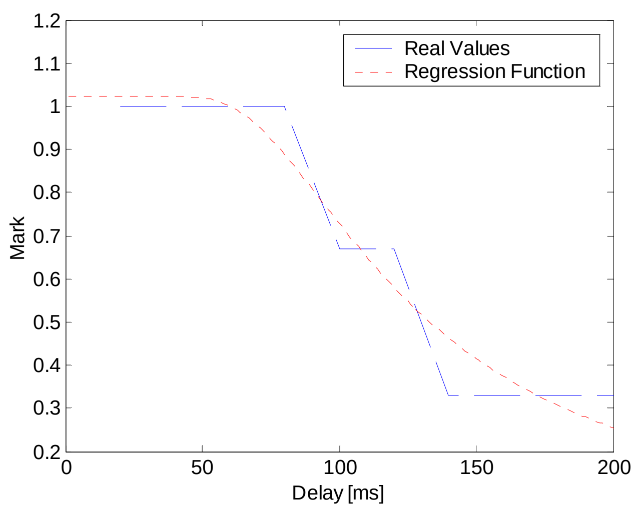

Figure 3-21 Show the subjective quality versus a regression function for Transfer File of 64 kbps.

The next equation shows the function of regression that has been found to the Whiteboard with a bandwidth of 384 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^{d}} \quad \textbf{(3-10)}$$

The Coefficient Data are:

a = 1.0221992
b = 0.99313688
c = 36356.358
d = -2.2372998

Standard Error: 0.0916969

Correlation Coefficient: 0.9714986

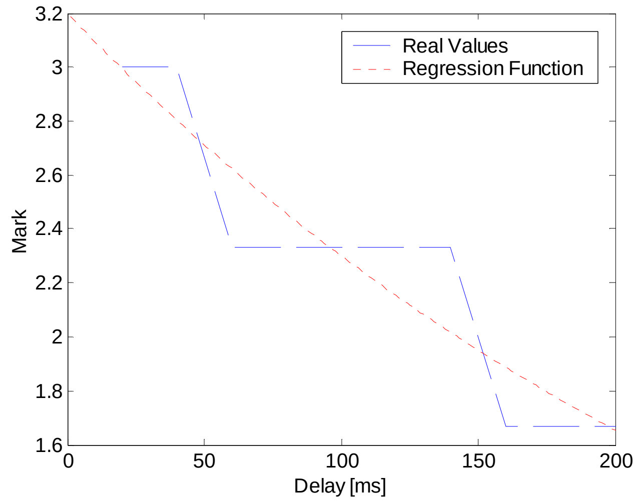

Figure 3-22 Show the subjective quality versus a regression function for Transfer File of 128 kbps.

The next equation shows the function of regression that has been found to the Transfer File with a bandwidth of 128 kbps. The name of the function is Exponential Fit.

$$f(x) = \mathrm{a}\exp^{\mathrm{bx}} \qquad \textbf{(3-11)}$$

a = 3.1989809
b = -0.0032964864

Standard Error: 0.2028190
Correlation Coefficient: 0.9208162

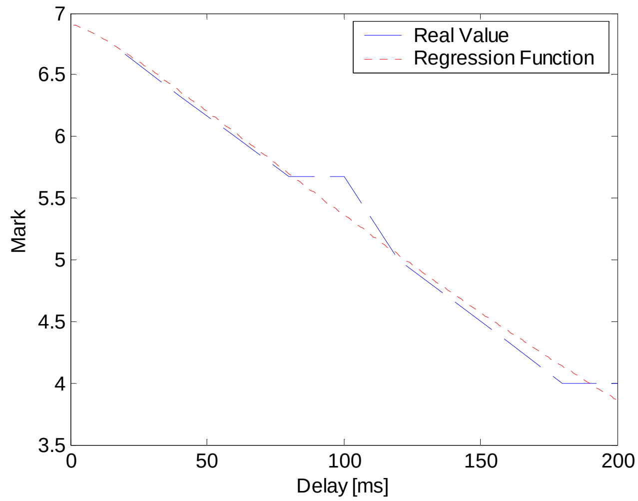

Figure 3-23 Show the subjective quality versus a regression function for transfer file of 384 kbps

The next equation shows the function of regression that has been found to the Transfer File with a bandwidth of 384 kbps. The name of the function is a linear fit.

$$f(x) = a + bx \tag{3-12}$$

The values of the coefficient data are:
a = 6.9793333
b = -0.015866667

Standard Error: 0.1405584
Correlation Coefficient: 0.9906212

In this service according to MOS measurements is very influenced by the delay and the bandwidth of the service. The best case is the only acceptable, because the rest has a very bad mark, the value is under 3.

Table 3-5 Average values of perceptual quality of Share Desktop

64 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	1,67	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
128 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	5,00	4,33	4,00	3,00	2,33	1,33	1,00	1,00	1,00	1,00
384 kbps										
Delay (ms)	20	40	60	80	100	120	140	160	180	200
Mark	8,67	8,33	8,00	7,33	6,67	5,67	5,00	4,67	4,67	4,67

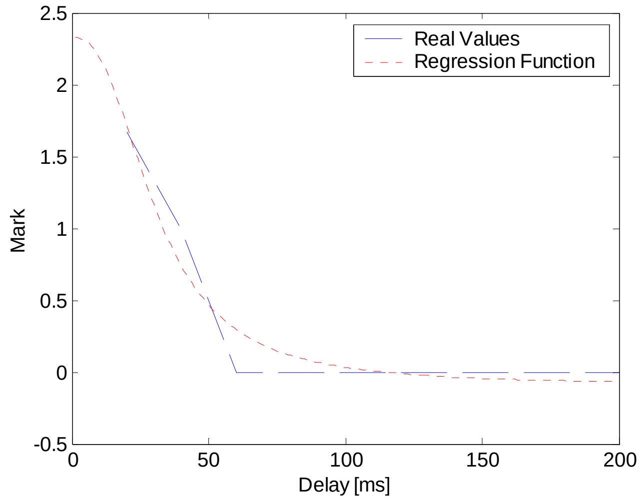

Figure 3-24 Show the subjective quality versus a regression function for Share Desktop of 64 kbps.

The next equation shows the function of regression that has been found to the Share Desktop with a bandwidth of 64 kbps. The name of the function is Weibull Model.

$$f(x) = \mathrm{a} - \mathrm{bexp}^{-\mathrm{cx}^{\mathrm{d}}} \quad \textbf{(3-13)}$$

Where the value of the coefficient data:

a = -0.090339749
b = 0.00019310307
c = 2.3302338
d = -2.4948962Standard Error: 0.1750782
Correlation Coefficient: 0.9696443

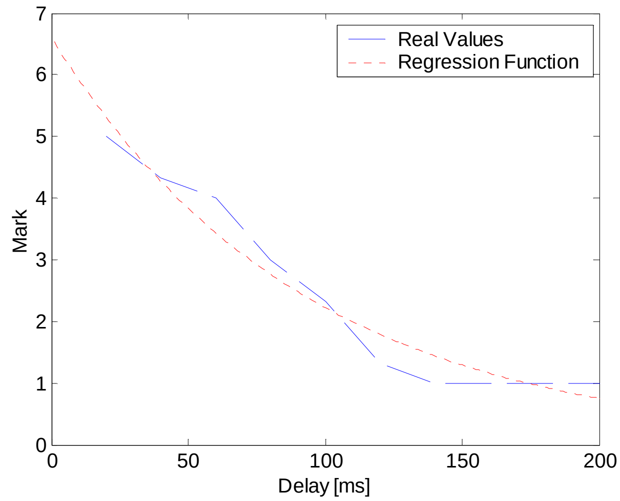

Figure 3-25 Show the subjective quality versus a regression function for Share Desktop of 128 kbps.

The next equation shows the function of regression that has been found to the Share Desktop with a bandwidth of 128 kbps. The name of the function is a exponential.

$$f(x) = \mathrm{aexp}^{\mathrm{bx}} \quad \textbf{(}3\text{-}14\textbf{)}$$

The values of the Coefficient Data are:
a = 6.6058433
b = -0.010883588

Standard Error: 0.3499898
Correlation Coefficient: 0.9778764

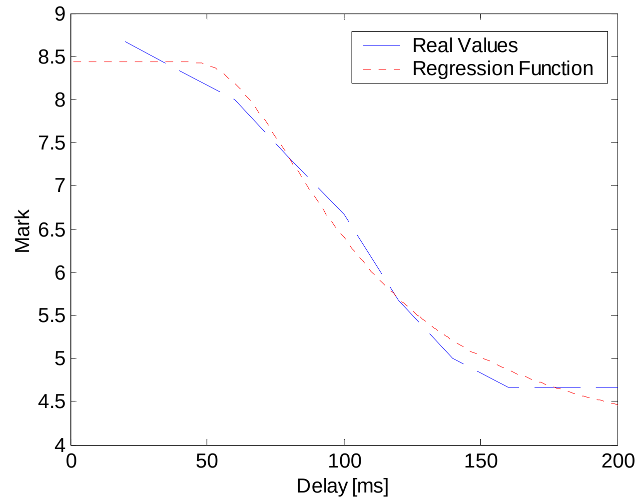

Figure 3-26 Show the subjective quality versus a regression function for Share Desktop of 384 kbps

The next equation shows the function of regression that has been found to the Share Desktop with a bandwidth of 384 kbps. The name of the function is Weibull Model.

$$f(x) = a - b\exp^{-cx^d} \quad \textbf{(3-15)}$$

The coefficient of the data are:
a = 8.4433095
b = 4.5821855
c = 84418.707
d = -2.5095521

Standard Error: 0.2231988
Correlation Coefficient: 0.9937104

Share Desktop service has a high influence of delay, it is possible to see that the value fall down is all the case. The best case and the acceptable according to the MOS measurement has a bandwidth of velocity.

Chapter 4. UMTS network

4.1 UMTS channels

UMTS network protocol stack is divided in different layers. This part of the work is dedicated to study especially the layer 1 and 2 of UMTS radio access network as this has the biggest influence on the perceived user quality.

The layer two is named data link layer and is formed by two sublayers named Medium Acces Control (MAC) and Radio Link Control (RLC), in addition there are two dependent service (in the user plane): Packet Data Convergence Protocol (PDCP) and Broadcast/Multicast Control Protocol (BMC). Control plane is represented by the radio resource control (RRC) protocol.

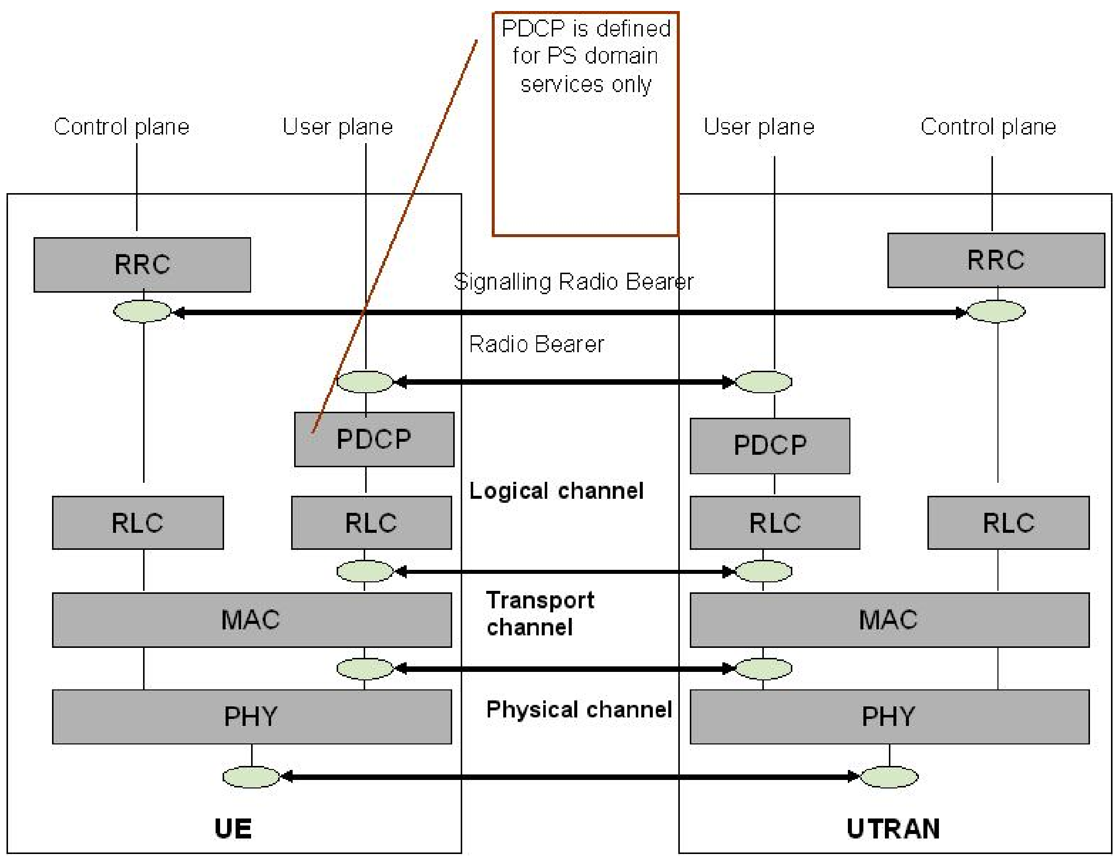

Figure 4-1 UMTS layers **[10]**

The idea of a layer structure is that each layer offers a service to the superior layer. For example the Physical Layer (Layer 1) offers its service to MAC layer and the MAC layer offers its service to RCL layer.

4.1.1 The Medium Access Control Protocol

One of the functions of the MAC layer is mapping of the logical channels to transport channels. It is responsible to select the correct transport format for each transport channel.

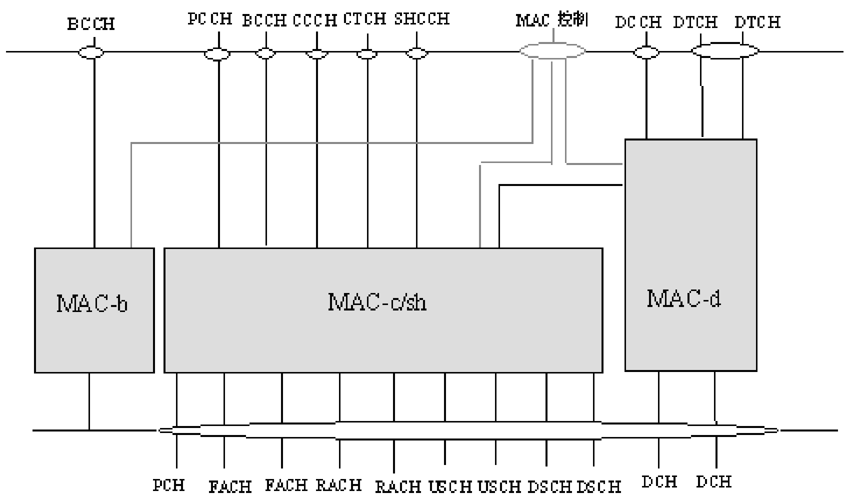

Figure 4-2 MAC Layer

[9]

The MAC layer Architutecture is formed by:

MAC-b supports broadcast channels

MAC-c/sh supports the common and shared channels including the paging channel, common packet channel, downlink shared channel forward and random access channels.

MAC-d supports dedicated transport channels

4.1.1.1 MAC functions

- Mapping of logical channel onto appropriate transport channel

- Selection of the appropriate Transport Format for each transport channel, depending of the instantaneous source rate.
- Priority handling between data flows of one UE. Each flow can have a different priority level.
- Traffic volume monitoring. It has the information of the data on the RLC buffer.
- Dynamic Transport channel type switching. Change the channel between common and dedicated transport channels based in a decision of RRC.
- Multiplexing/demultiplexing of higher layer PDU into/from transport blocks delivered to/from the physical layer on common transport channel.

Physical layer offers to the MAC three types of transport channels (TrCH): dedicated, common and shared. Dedicated channel (DCH) is allocated to the one single user for the whole communication. It is an bidirectional (UL+DL) channel. Common channels are the channels common for all users. In uplink there is an random access channel (RACH) used for an initial access and also for the transport of low data rate user data (up to 32 kbps is recommended). In downlink there is a forward access channel (FACH) used for answering the initial access requests. Via this channel also the user data with lower data rates can be sent (up to 32kbps recommended). These channels are not power controlled and are suitable for low data rate bursty traffic, as they do not remain reserved and resource negotiation is necessary for every burst sent. Shared channels are reserved, but can be used by more users – they are time divided. From the point of view of the user they are dedicated (read by only one user), but they are shared from the point of view of the network. Shared channels are downlink only or frequency division duplex (FDD) UMTS that is commonly used nowdays. Downlink shared channel (DSCH) and High Speed Downlink Schared Channel (HS-DSCH) were proposed to allow the assymetrical traffic with higher bandwidth consumption in the downlink direction. HS-DSCH is defined from Rel-5 and is characterized by higher capacity – up to 11Mbps per cell. It also has shorter interleaving period (TTI) allowing for faster reactions.

4.2 Radio link Architecture

The layer 3 of UTRAN is Radio Link Control. This layer offers segmentation and retransmission service for both user and control data. The RLC is configured by RRC, its can be operate in one of this three modes:

Transparent mode: In this mode no overhead is added to higher layer data. The PDU that result erroneous are marked or discarded depend of the configuration. No data segmentation is applied. This mode is used only for the circuit switched (CS) data.

Unacknowledged mode: In this mode no retransmission protocol is used. The PDU that are erroneous are marked or discarded depending on the configuration.

All the PDUs have a sequence number to check the integrity. The PDUs that are not transmitted in the time are discarded.

Acknowledged mode: In this mode the communication is always answered by a confirmation (ACK) that the PDU envoy is correct. In the case that it is not possible to transmit the PDU correctly due to the timer or maximum number of retransmission, it is discarded upper layers are informed. The retransmission always implies a delay. It is possible to control the maximum delay or maximum time of retransmission by the RRC parameters that are configured by the maximum time of retransmission allowed for guaranteed class of services.

4.3 RLC Functions.

This layer has defined the following functions:

Segmentation and reassembly: This function segment variable size of the PDU of the upper layers to a smaller RLC payload units. One RLC unit PDU carries one Packet Unit.

Concatenation: If the context of the RLC SDU is not a fix number of RLC PU, the first segment of the next RLC SDU may be input into the RLC PU in the concatenation with the last segment of the previous RLC SDU.

Padding: It is not possible to concatenation the PDU. The transmission of the PDU needs a complete RLC PDU. The remainder data are bit of padding.

Transfer user data: RLC support transparent mode, Unacknowledged and Acknowledged mode.

Error Correction: By means of the retransmissions performed in acknowledged mode.

Duplicate Detection: Detects PDUs that are duplicated, and serve only one PDU to the upper layer.

Flow Control: The RLC receiver can check the rate of the transmission.

Sequence number check: Each PDU has a sequence number to check it, in the reassembly process.

4.4 The Packet Data Convergence Protocol

The packet data convergence protocol (PDCP) is part of the layer 2 and exist only in the user plane and the service from Packet Switch (PS). It has the property of compressing the header of RTP/UDP/IP.

4.5 QoS in UMTS.

The UMTS network allows to have QoS. The table 4-1 show the 4 type of traffic class that UMTS supports.

Table 4-1 Type of traffic class for UMTS. **[16]**

Traffic class	Conversational class conversational RT	Streaming class streaming RT	Interactive class Interactive best effort	Background Background best effort
Fundamental characteristics	-Preserve time relation (variation) between information entities of the stream Conversational pattern (stringent and low delay)	-Preserve time relation (variation) between information entities of the stream (i.e. some but constant delay)	Request response pattern Preserve payload content	Destination is not expecting the data within a certain time Preserve payload content
Example of the application	-speech, video, …	-facsimile (NT) -streaming audio and video	- Web browsing	-background download of emails

4.6 Radio Access Bearers (RABs) for UMTS

Radio Access Bearers are the service offered by UTRAN (UMTS Terrestrial Radio Access Network). In 3GPP there is specification [13] defining some RABs for conformance terminal testing. Further example RABs can be found in technical report [25.993]. Of course, the operators have the possibility to define and configure their own RABs. In following the possible suitable RABs will be chosen for the studied services of NetMeeting.

The next tables show different defined RABs for conformance testing.

Table 4-2. Examples of Radio Access Bearers (RABs).

#	Traffic class [2]	SSD	Max. rate, kbps	CS/PS
1	Conversational	Speech	UL:12.2 DL:12.2	CS
2	Conversational	Speech	UL:10.2 DL:10.2	CS
3	Conversational	Speech	UL:7.95 DL:7.95	CS
4	Conversational	Speech	UL:7.4 DL:7.4	CS
5	Conversational	Speech	UL:6.7 DL:6.7	CS
6	Conversational	Speech	UL:5.9 DL:5.9	CS
7	Conversational	Speech	UL:5.15 DL:5.15	CS
8	Conversational	Speech	UL:4.75 DL:4.75	CS
9	Conversational	Unknown	UL:28.8 DL:28.8	CS
10	Conversational	Unknown	UL:64 DL:64	CS
11	Conversational	Unknown	UL:32 DL:32	CS
12	Conversational	Unknown	UL:8 DL:8	PS
13	Conversational	Unknown	UL:16 DL:16	PS
14	Streaming	Unknown	UL:14.4 DL:14.4	CS
15	Streaming	Unknown	UL:28.8 DL:28.8	CS
16	Streaming	Unknown	UL:57.6 DL:57.6	CS
17	Streaming	Unknown	UL:0 DL:64	CS
18	Streaming	Unknown	UL:16 DL:64	PS
19	Streaming	Unknown	UL:64 DL:0	CS
20	Streaming	Unknown	UL:8 DL:16	PS
21	Streaming	Unknown	UL:8 DL:32	PS
22	Streaming	Unknown	UL:16 DL:64	PS
23	Streaming	Unknown	UL:32 DL:256	PS
24	Void			
25	Streaming	Unknown	UL:16 DL:128	PS
26	Void			
27	Void			
28	Interactive or Background	N/A	UL:32 DL:8	PS
29	Interactive or Background	N/A	UL:8 DL:8	PS
30	Interactive or Background	N/A	UL:16 DL:16	PS

31	Interactive or Background	N/A	UL:32 DL:32	PS
32	Interactive or Background	N/A	UL:64 DL:8	PS
33	Interactive or Background	N/A	UL:32 DL:64	PS
34	Interactive or Background	N/A	UL:64 DL:64	PS
35	Interactive or Background	N/A	UL:64 DL:128	PS
36	Interactive or Background	N/A	UL:128 DL:128	PS
37	Interactive or Background	N/A	UL:64 DL:384	PS
38	Interactive or Background	N/A	UL:128 DL:384	PS
39	Interactive or Background	N/A	UL:384 DL:384	PS
40	Interactive or Background	N/A	UL:64 DL:2048	PS
41	Interactive or Background	N/A	UL:128 DL:2048	PS
42	Interactive or Background	N/A	UL:384 DL:2048	PS
43	Interactive or Background	N/A	UL:64 DL:256	PS
44	Interactive or Background	N/A	UL:0 DL:32	PS
45	Interactive or Background	N/A	UL:32 DL: 0	PS
46	Interactive or Background	N/A	UL:0 DL:0	PS
47	Interactive or Background	N/A	UL:64 DL:144	PS
48	Interactive or Background	N/A	UL:144 DL:144	PS
49	Interactive or Background	N/A	UL:128 DL:32	PS
50	Streaming	Unknown	UL:16 DL:16	PS
51	Streaming	Unknown	UL:16 DL:32	PS
52	Interactive or Background	N/A	UL:16 DL:32	PS
53	Interactive or Background	N/A	UL:16 DL:64	PS

54	Interactive or Background	N/A	UL:16 DL:128	PS
55	Streaming	Unknown	UL:128 DL:16	PS
56	Conversational	Speech	UL:(12.65 8.85 6.6) DL:(12.65 8.85 6.6)	CS
57	Conversational	Unknown	UL:42.8 DL:42.8	PS
58	Interactive or Background	N/A	UL:128 DL:64	PS
59	Interactive or Background	N/A	UL:384 DL:64	PS

4.6.1 Calculation of maximum number of bits per TTI.

$$\frac{1}{code\_rate}(TF\_size + TBs \quad \cdot CRC) + TAIL \quad \cdot ceil\left(\frac{TF\_size + TBs \quad CRC}{code\_block\_size}\right)$$

The turbo coding rate has a value of $1\!\!\diagup\!_3$.

And therefore the value for 1/code_rate is 3 for all our RABs as the turbo code is usually used for packet services. The value of TF_SIZE that is the value of the biggest TFx where x is the highest number that is possible to see in the table 4-3, in this example is 4, and the value of TF_size is 8 x328. The value of TBs is the first part of this operation 8x328 (8), and the CRC is the number of CRC bits as canbe seen from the table. The TAIL value is a constant and is 12 for turbo coding, and the code_block_size is 5114 bits. The result of the operation is gotten the value of maximum number of bits.

4.6.2 Video

There are no RABs defined in [12] and [13] for video over PS domain that NetMeeting needs. The NetMeeting video needs at least about 128 kbps in each direction, if ran with CIF resolution (352x288), that is its default setting. For QCIF the bandwidth needed should be lower, but anyhow, there are no conversational class PS RABs with higher data rates defined up to now. It is not possible to use other traffic class because the video is highly delay sensitive as can be seen from the MOS measurement.

This is the reason why I do not recommend touse Videoconference offered by NetMeeting applications when running over UMTS. In the actually the UMTS mobile offers Videoconference using only 64 kbps in each direction of the communication over CS domain. In following I will try define the transport channel parameters for a new RAB for the Video over PS domain. For video I propose to use the data rate 128kbps in each direction and to use minimum interleaving period of 10ms. Video needs to be multiplexed onto the DCH channel as it has a constant bit rate nature and is symmetrical. Unacknowledged mode is proposed to minimize the delay.

The Table 4-3 show the value of the RAB that has been selected to the uplink:

Table 4-3 Transport channel parameters for Interactive or background / UL:128 kbps / PS RAB

Higher layer	RAB/Signalling RB		RAB
RLC	Logical channel type		DTCH
	RLC mode		UM
	Payload sizes, bit		320
	Max data rate, bps		128000
	AMD PDU header, bit		8
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		328
	TFS	TF0, bits	0x328
		TF1, bits	1x328
		TF2, bits	2x328
		TF3, bits	4 x328
		TF4, bits	8 x328
	TTI, ms		10
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		8272
	Uplink: Max number of bits/radio frame before rate matching		4230
	RM attribute		120-160

The Table 4-4 shows the value of the RAB of the download that has been selected:

Transport channel parameters for Interactive or background / DL:128 kbps / PS RAB

Table 4-4 Transport channel parameters for Interactive or background / DL:128 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		UM
	Payload sizes, bit		320
	Max data rate, bps		128000
	AMD PDU header, bit		8
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		328
	TFS	TF0, bits	0x328
		TF1, bits	1x328
		TF2, bits	2x328
		TF3, bits	4 x328
		TF4, bits	8 x328
	TTI, ms		10
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		8272
	Uplink: Max number of bits/radio frame before rate matching		4230
	RM attribute		120-160

4.6.3 Audio

The fastest conversational/speech class for PS domain has a speed of 42.8 kbps in UL and DL of 42.8 kbps and can be found in [14] and in the Table 4-2 below. This is speech encoded by AMR (Adaptive Multirate) codecPlease note, that this RAB is defined first in release 5 of UMTS standard to work over its IMS architecture. There are no defined mode for speech by the default speech codec of NetMeeting. One could force netmeeting to use AMR. Is not recommendable to use speech of NetMeeting codecs, It is better to use the phone communication over CS domain, because the rate is from 4.75 to 12.2 kbits. The reason for the big difference between the bandwidth needed for AMR over PS and CS domain is the necessity of the IP/UDP/RTP header attached to each AMR packet. The header is bigger than the small well-compressed speech packets. UMTS offers the possibility to use the header compression as for example Robust Header Compression (ROHC), but it is difficult to specify the RAB with reduced data rate accordingly as ROHC algorithm produces variable bit rate stream due to the updates of header.

NetMeeting needs for acceptable communication even more - 64 kbps of bandwidth and actually UMTS do not have defined such RABs. It could be done, but it is not really economical and recommendable. In PS domain there is also more difficult to full fill the QoS requirements that are very high in case of speech (high delay and jitter sensitivity as can be seen from the MOS measurements).

The RAB that has been proposed to the audio service for the uplink is in the table 4-5:

Table 4-5 Transport channel parameters for conversational / UL:64 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		UM
	Payload sizes, bit		320
	Max data rate, bps		64000
	AMD PDU header, bit		8
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		328
	TFS	TF0, bits	0x328
		TF1, bits	1x328
		TF2, bits	2x328
		TF3, bits	3x328
		TF4, bits	4x328
	TTI, ms		10
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		4114
	Uplink: Max number of bits/radio frame before rate matching		2118
	RM attribute		130-170

The RAB that has been proposed to the audio service for the download is in the table 4-6:

Table 4-6 Transport channel parameters for conversational / DL:64 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		UM
	Payload sizes, bit		320
	Max data rate, bps		64000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		328
	TFS	TF0, bits	0x328
		TF1, bits	1x328
		TF2, bits	2x328
		TF3, bits	3x328
		TF4, bits	4x328
	TTI, ms		10
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		4114
	Uplink: Max number of bits/radio frame before rate matching		2118
	RM attribute		130-170

4.6.4 WhiteBoard

For Whiteboard the RAB found is presented below and it can be used. According to the MOS measurements Whiteboard is not so much delay sensitive and it has more the character of Interactive/Background class.

The next table show the characteristics of the RAB that is proposed to whiteboard. The data rate of 64 kbps according of the MOS is enough of course, is the data rate would be faster, It would be ok, too.

The table with Transport channels parameters has been extracted of UMTS specifications **Error! Reference source not found.**]. The physical channel parameters as well as the Transport format combination set together with signalling radio bearer (SRB) has to be defined additionally. SRB stays the same as described in [15] for all RABs.

The RAB that has been proposed to the audio service for the uplink is in the table 4-7 and the download side is in the table 4-8:

Table 4-7 Transport channel parameters for Interactive or background / UL:64 PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		64000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		FACH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
		TF2, bits	2x336
		TF3, bits	3x336
		TF4, bits	4x336
	TTI, ms		20
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		4236
	Uplink: Max number of bits/radio frame before rate matching		2118
	RM attribute		130-170

Table 4-8 Transport channel parameters for Interactive or background DL:8 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		8000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		FACH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
	TTI, ms		40
	Coding type		TC (alt. CC 1/3)
	CRC, bit		16
	Max number of bits/TTI after channel coding		1068 (alt. 1080)
	RM attribute		135-175

4.6.5 Transfer File

In the MOS the acceptable RAB has to be a 384 kbps. In real life it is possible to download from a server a file of 2 MB and the bandwidth and the delay is acceptable.

I proposed the RAB that in indicated in MOS, and is presented in table 4-9 and 4-10.

The values has been taken from [12]

Table 4-9 Transport channel parameters for Interactive or background / UL:384 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		384000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
		TF2, bits	2x336
		TF3, bits	4 x336
		TF4, bits	8 x336
		TF5, bits	12x336
		TF6, bits	16x336(alt. N/A)
		TF7, bits	20x336(alt. N/A)
		TF8, bits	24 x336 (alt. N/A)
	TTI, ms		20 (alt. 10)
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		25368
	Uplink: Max number of bits/radio frame before rate matching		12684
	RM attribute		110-150

Table 4-10 Transport channel parameters for Interactive or background / DL:64 kbps / PS RAB

Higher Layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		64000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
		TF2, bits	2x336
		TF3, bits	3x336
		TF4, bits	4x336
	TTI, ms		20
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		4236
	RM attribute		130-170

4.6.6. Chat

There are no MOS measurements to have a objective value of the RAB that is more precise. It has been selected the next the characteristics of the chat is a very poor requeriment of bandwidth and delay. This is the reason that has been selected the next RAB, in the table 4-11 and table 4-12 show the RAB for the uplink and download side.

Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps
SRBs for DCCH:

Table 4-11 Transport channel parameters for Interactive or background / UL:8 kbps / PS RAB

Higher Layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		8000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		FACH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
	TTI, ms		40
	Coding type		CC 1/3 (alt. TC)
	CRC, bit		16
	Max number of bits/TTI after channel coding		1080 (alt. 1068)
	Uplink: Max number of bits/radio frame before rate matching		270 (alt. 267)
	RM attribute		135-175

Table 4-12 Transport channel parameters for Interactive or background / DL:8 kbps / PS RAB

Higher layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		8000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		FACH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
	TTI, ms		40
	Coding type		CC 1/3 (alt. TC)
	CRC, bit		16
	Max number of bits/TTI after channel coding		1080 (alt. 1068)
	RM attribute		135-175

4.6.7 Desktop Sharing.

The service that offers NetMeeting in this case it possible to thing that has the same requirements that File Transfer. The problem is in the Desktop Sharing is normal that one person speak and at the same time move the information in the Desktop. This is important, for example, in a conference that the person that show a virtual presentation in the other part of the world do not give more information to the audience that they can see at the moment. For this reason is very important the requirement of QoS in this service. How it has been commented before the Desktop sharing work over TCP protocol and this protocol has a ack answers to the packet of data. This traffic is about 10 % of the communication.

The RAB that is defined is the table 4-13 and table 4-14 for uplink and download.

Table 4-13 Transport channel parameters for stream / UL:384 kbps / PS RAB

Higher Layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		384000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
		TF2, bits	2x336
		TF3, bits	4 x336
		TF4, bits	8 x336
		TF5, bits	12x336
		TF6, bits	16x336(alt. N/A)
		TF7, bits	20x336(alt. N/A)
		TF8, bits	24 x336 (alt. N/A)
	TTI, ms		20 (alt. 10)
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		25368
	Uplink: Max number of bits/radio frame before rate matching		12684
	RM attribute		110-150

Table 4-14 Transport channel parameters for stream / DL:64 kbps / PS RAB

Higher Layer	RAB/Signalling RB		**RAB**
RLC	Logical channel type		DTCH
	RLC mode		AM
	Payload sizes, bit		320
	Max data rate, bps		64000
	AMD PDU header, bit		16
MAC	MAC header, bit		0
	MAC multiplexing		N/A
Layer 1	TrCH type		DCH
	TB sizes, bit		336
	TFS	TF0, bits	0x336
		TF1, bits	1x336
		TF2, bits	2x336
		TF3, bits	3x336
		TF4, bits	4x336
	TTI, ms		20
	Coding type		TC
	CRC, bit		16
	Max number of bits/TTI after channel coding		4236
	RM attribute		130-170

Chapter 5. Conclusions

NetMeeting is a very complete program constructed under standard of the ITU. It is a Microsoft program. This program has a lot of services and is very easy to use. All the services can be used over internet network or private IP network like UMTS. It is true that has been connection to ISDN network with a gateway.

The interface is really easy to use and the quality of the service almost on all the case is excellent.

Other important characteristic is that it is a free application. If you have windows installed you can use it. The NetMeeting program is a program that is installed by default in windows system from Windows ME to Windows XP. The competence of this program like pcAnywhere 10.0 is more complete and simpler to use. The Pcanywhere has authentication and encryption (AES 256), can have too a file transfer and chat and other service. It is a really completely commercial program. All the protocols that have been implemented in pcAnyWhere are property of Symantec. This is a perfect program for a business or factory. There are two versions, and the price of the cheapest one is 200 €. There are other free applications like vnc but there are studies that say that it has a spyware. This is the reason that I think that for a home user NetMeeting is a really good tool.

The MOS measurements were a really big surprise, the most important are the big requirements of bandwidth of the users, a example of this is a speech it is possible to see that in 128 kbps the delay of the communication has a very big influence on it. The delay in the mobile networks like UMTS is always bigger than the fixed networks and this is the reason that the delay is important and in the graphs it is shown how it affects the communication from the user point of view. Higher data rates are needed than those acceptable for UMTS.

In the last release of UMTS network there are not defined channels that would be needed to transmit data over UMTS network, but of course, it possible to define a new RAB with the characteristics that are necessary. The channels have been defined, but it has been justified in the document that it is better not to use the service that is provided by NetMeeting. The principal reason is that it is more efficient to use the service provided over UMTS CS domain. It has less consumption of resources in the UMTS. The service that has been spoken about is video and speech. The UMTS network has defined a class of service for this traffic. The class is conversational and in the case of speech a lot of codecs with different data rate were defined and each 20 ms the network can change depending of the state of it. The AMR codec goes from 4.75 to 12.2 kbps if sent over CS domain without the need of RTP/UDP/TCP packetization. NetMeeting according to the MOS needs 64 kbps in full-duplex conversation. Of course, it only needs 32 kbps in one way, but this is the double of the best quality of the codec defined in UTMTS.

There is also defined a class for video, the big difference with respect for example to GSM (It only has a 9.6 kbps to transmit data in CS) is with UMTS is possible to transmit video in real time.

The latest cellular mobiles have a camera that allows to have a video conference instead of having a simple voice conference. Using a data transmission rate of 64 kbps in each direction provides a good relation between used resources and the quality. The best trade off has to be found to achieve the optimum quality of the image and the speech transmission at the same time. The price of video conference using a mobile phone (CS domain) in spain is about 1.5 €/min. For instance: The price of the download volume of 1 megabyte in the best (cheapest) case is 0.5 €. The quality of speech transmission in a GSM network is always better than using NetMeeting. In the case of speech it is possible to compare it also according to the price of the megabyte of data and the quality of the codec. It is important to remark that the quality of the AMR codec it better than the codes used in GMS network and it possible than in a future with best codec that has been implemented and optimised the quality will be the same, but at the moment this does not occur. There are other programs like skype that allow to have a conference and the quality of the conversation is absolute better than NetMeeting.

The RAB that has been proposed for speech is a conversational with 128 kbps in full duplex channel with an UM RLC mode, the video works over UDP protocol and in this type of protocol there is not guaranteed that the info arrives to the destination. All the process that checks that the information was correctly received and could ask for a retransmission takes a requirements of time and in real time services this time is not acceptable. It is important to remark that the information that UDP sends to the upper layer is only the one without an error. The information that has error has been discarded and the packet is considered to be lost. The protocol in application layer has been designed for the case where only few data has been lost that does not affect too much the communication.

In both cases - for speech and for video - there are only 10 ms of interleaving period, the reason is the same – the delay. TTI is the time needed for receive/send the block of data. It is not the time that the system wait to process it. The total time is always even bigger.

This is the reason why I would propose not to use these services (video and speech) that offers NetMeeting and instead to work with the mobile CS services of UMTS.

If we want to multiplex speech and video offered by NetMeeting on an PS domain RAB in UMTS anyhow, we will for sure need the DCH transport channel.

The reason is the high symetricity of these two services as well as the more or less constant requirement on bandwidth and delay.

The other service possible to use is whiteboard. This is a service with really low requeriments on QoS, comparing with the other services that NetMeeting offers. In the worst case of the MOS measurements with a value of 64 kbps and delay of 200 ms it has a mark of 6.75 and it is possible to see that with the best value the mark goes with the same characteristics to 8.3. ThereforeI proposed a 64 kbps channels because it is sufficient for this service.

This is a service that only is implemented in a very few programs. GnomeNeeting is also one of them. The RAB selected is with a DTCH as a logical channel. It can be mapped on FACH, DCH, DSCH. As RLC mode AM was chosen. The reason is that in this case it is more important that the info arrives without any error with complete integrity. And in transport layer this protocol works over TCP and it detects a loss packet or corrupted packet, there are a retransmissions of the data in the transport layer, not only at RLC. The TTI period is 20 ms, because it is important for the integrity of the data (bigger interleaving period increases the probability of correct detection).

The file transfer service has a requirement of bandwidth according to the MOS measurements really higher. It is true that the size of the file that had been making the MOS measurement was very big, about 2 MB. Comparing the results with other technologies for example modem of 33.6 kbps is sufficient to work. And with this type of modem a lot of people are connecting to internet and working every day, of course that the medium of propagation is not the air and the delay of the communication is better than air communication. (This type of modem has a compression of the header of TCP, too). This is the reason why I supposed that the requirement of bandwidth in this case is excessive and I proposed a 64 or 128 kbps of download although according to the MOS measurements it is 384 kbps. This is also caused by the fact, that the file transfer can be normally used as a background service, the people can use the time of downloading to perform other actions, to browse etc. In our MOS measurements we let the people wait to download the file because we wanted to see the real limit for their patiency.

The RAB proposed has a logical channel of DTCH (It is a traffic channel, no control o signalling channel) and the reason is the same as before. It could be mapped on DCH, DSCH, HS-DSCH or FACH/RACH (in this case has been selected DCH transport channel). The mode that has been selected in RLC is AM. The reason is that is very important the integrity of the data and the delay is not really critical. And the protocol in the layer 4 is TCP. This has a retransmit of corrupted or loosed data. The interleaving time is 20 ms, but could also be 40ms. The reason is the same is important to secure the integrity of the data.

The instant messaging service (Chat) is the service that required less QoS that all the service that has been explained. There are no requirements of delay in real time communication, because in speech the delay what we talked about had a magnitude of ms, and instant message it is possible to have the delay of seconds and it is completely acceptable to the users. The reason is a perception of the users, they can spend 3 seconds writing a phrase. If these message do not arrive in 1 second more is not so much important. The requirements of bandwidth for this service are really low. It has been talked about 0.5-5 kbps. This is the principal reason that has been proposed a RAB with 8 kbps.

The RAB proposed is explained in the document, the logical channel selected is DTCH and it can be mapped on FACH transport channel in this case. The reason because that has been selected a FACH channel is the very low data rate of data that instant messages have. In this service it is important the integrity if the data and not the time that the information arrives. There is very important too the integrity of the data and this is the reason that has been selected AM mode in RLC and a TTI of 20 ms.

The Desktop sharing is a service that required a special QoS. The reason is that it can be used to make a virtual meeting for example and it is very important that the info goes in real time with the voice/speech. It is no occurs and the voice goes faster than the information of the Desktop the audience can loose a lot of information because they are not looking while the the person is speaking. This is the reason why has been selected a streaming class of traffic. And how it is possible to see the table that has been defined is for 384 kbps to achieve the perfect quality, but 128kbps would be still acceptable and more feasible.

In this case it is important that the info arrives in time, but it is important too, that the integrity of the data is assured. The reason is this service works with TCP in layer 4 and if there are any errors of packet has been lost in the way, the TCP protocol will make a retransmission of the data. And this costs a time. That has been explained in the before, therefore a streaming class has been selected, and in RLC mode has been selected an AM. To ensure the integrity of the data that will be transmitted and at the same time to minimize the delay. The DTCH is mapped to DCH channel, the reason is the requirement of datarate. The other channel cannot support it – only the HS-DSCH could be suitable as well, but it is available first in the release 5 of the UMTS standard.

In general the UMTS network has a flexibility to select or create a new RAB that the users can need. But it is important to know that all the RABs that have been selected for us can be denegate for the UMTS if is not possible to offer the QoS required due to the high load or bad channel conditions.

The UMTS network is a good channel to transmit the data and in the case that this is not recommendable the same network can offer to us different tools that facility the same service with better quality.

Chapter 6. Abbreviations

AM	Acknowledged mode
BCH	Broadcast channel
BMC	Broadcast/Multicast Control Protocol
BSD	Berkeley Software Distribution
CPCH	Common Packet Channel
CS	Circuit Switched
DCCH	Dedicated control channel
DCH	Dedicated channels
DL	Downlink
DPCCH	Dedicated Physical Control Channel
DPCH	Dedicated Physical Channel
DSCH	Downlink Shared channel
DPDCH	Dedicated Physical Data Channel
DTCH	Dedicated traffic channel
FACH	Forward link access channel
FDD	Frequency division duplex
GNU	General Public License
GOB	Group of blocks.
GPL	GNU Public License
HS-DSCH	High speed Download Channel
IP	Internet Protocol
ITU	International Telecommunication Unit
MAC	Medium Access Control
MS	Mili seconds
MOS	Mean Opinion Score
MTU	Maximum Transfer Unit.
NAT	Network Address Translator
OSI	Open Systems Interconnection
PCH	Paging channel
PCM	Pulse Code Modulation.
PDCP	Packet Data Convergence Protocol
PDU	Protocol data Unit.
PHY	Physical Layer
PS	Packet Switched
PU	Payload Unit.
QoS	Quality of Service
RAB	Radio Access Bearer

RB	Radio Bearer
RACH	Random Access channel
ROCH	Robust Header Compression.
RLC	Radio Link Control
RRC	Radio Resource Control.
RTP	Real Time Transport Protocol
SDU	Service data unit.
TC	Turbo Coding
TCP	Transmision Control Protocol
TFCI	Transport Format Combination Indicator
TFCS	Transport Format Combination Set
TTI	Interleaving period.
UDP	User Datagram Protocol
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UM	Unacknowledged mode

Chapter 7. Bibliography

[1] Microsoft Corporation, NetMeeting Architecture, Chapter 8. In "*Microsoft® Windows® NetMeeting® 3 Resource Kit" pages* 155-160, Microsoft Corporation (1996-2001).

[2] Microsoft Corporation, Understanding the T.120 Standard, Chapter 10. In "*Microsoft® Windows® NetMeeting® 3 Resource Kit" pages* 167-176, Microsoft Corporation (1996-2001).

[3] Microsoft Corporation, NetMeeting Architecture, Chapter 8. In "*Microsoft® Windows® NetMeeting® 3 Resource Kit" pages* 155-160, Microsoft Corporation (1996-2001).

[4] http://www.ietf.org/rfc.html

[5] www.ethereal.com

[6] http://www.itu.int/rec/recommendation.asp?type=items&lang=E&parent=T-REC-H.245-200302-S

[7] http://info.iet.unipi.it/~luigi/ip_dummynet/

[8] Microsoft Corporation, Understanding H.323 Standard, Chapter 11. In "*Microsoft® Windows® NetMeeting® 3 Resource Kit" pages* 177 to 186, Microsoft Corporation (1996-2001).

[9] TONG Zhen Li, LI Ying Liang, ZHENG Jian Hong, MAC realization scheme, Chapter 3, in "*The Media Access Control of Terminals In TD_SCDMA System*", pages 2-6, www.tdscdma-forum.org

[10] T www.3GPP.com, "Radio interface protocol architecture", Cap 5, *in 3GPP TS 25.301 V3.b.0 (2002-09), Technical Specification 3rd Generation Partnership Project; Technical Specification Group Terminals;* Radio Interface Protocol Architecture *(Release 1999)*, *pages 9-36. 3GPP, Valbonne – FRANCE (2002)*

[11] Microsoft Corporation, Understanding the H.323 Standard, Chapter 11. In "*Microsoft® Windows® NetMeeting® 3 Resource Kit" pages* 182, Microsoft Corporation (1996-2001).

[12] www.3GPP.com, "Examples of Radio Bearers and Signaling Radio Bearers for FDD", Cap 7, in *3GPP TR 25.993 V6.8.0 (2004-12), Technical Report 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Typical examples of Radio Access Bearers (RABs) and Radio Bearers*

(RBs) supported by Universal Terrestrial Radio Access (UTRA) (Release 6), pages 36-117. 3GPP, Valbonne – FRANCE (2004)

[13] www.3GPP.com, "Reference System Configurations", Cap 6, *in 3GPP TS 34.108 V3.15.0 (2004-03), Technical Specification 3rd Generation Partnership Project; Technical Specification Group Terminals; Common test environments for User Equipment (UE) conformance testing (Release 1999)*, *pages 24-384. 3GPP, Valbonne – FRANCE (2004)*

[14] www.3GPP.com, "Examples of Radio Bearers and Signaling Radio Bearers for FDD", Cap 5, in *3GPP TR 25.993 V6.8.0 (2004-12), Technical Report 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Typical examples of Radio Access Bearers (RABs) and Radio Bearers (RBs) supported by Universal Terrestrial Radio Access (UTRA) (Release 6), pages 26-28. 3GPP, Valbonne – FRANCE (2004)*

[15] http://www.ebicom.net/~dhyams/cvxpt.htm

[16] www.3GPP.com, "Examples of Radio Bearers and Signaling Radio Bearers for FDD", Cap 4, in 3GPP TR 25.993 V6.8.0 (2004-12), Technical Report 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Typical examples of Radio Access Bearers (RABs) and Radio Bearers (RBs) supported by Universal Terrestrial Radio Access (UTRA) (Release 6), pages 24-26. 3GPP, Valbonne – FRANCE (2004)

Escola Politècnica Superior de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

INSTITUT FÜR NACHRICHTENTECHNIK UND HOCHFREQUENZTECHNIK

ANNEXOS

Title: Netmeeting: Performance and optimization for wireless networks

Author: Eduardo Dijort Romagosa

Director: Olívia Némethová

Data: 25 de febrer de 2005

APPENDIX

APPENDIX A Code of the program.

This code is the code of the program extract.exe. comment in the point 3.1.4.
The code is written in c#.

```
using System;
using System.IO;

namespace extract
{
      /// <summary>
      /// Summary description for Class1.
      /// </summary>
      class Class1
      {
            /// <summary>
            /// The main entry point for the application.
            /// </summary>
            [STAThread]
            static void Main(string[] args)
            {

                  string contenido;
                  int posicion;
                  string copia;
                  //    int dime;
                  int pos;
                  string total="";
            StreamReader ar = new StreamReader("C:\\entrada.txt");
      StreamWriter escribe = new StreamWriter("C:\\salida2.txt");
                  try
                  {
                        while (true)
                        {

                              try
                              {
                                    contenido= ar.ReadLine();
                              }
                              catch
                              {
                  Console.WriteLine("Detectado fin de fichero");
                                    break;
                              }
                              /*Posicion 8 siempre es la misma*/

            if (contenido==null) break;
            posicion = contenido.IndexOf("Total Length: ");
                        if (posicion!=-1)
                              {

      copia=contenido.Substring(posicion+13);
                                    total = " "+copia;
                                    escribe.Write(" "+total+" ");
```

```
                                escribe.WriteLine();
                        }
posicion = contenido.IndexOf("No.     Time        Source");
                        if (posicion==0)
                        {
                                try
                                {
                                        contenido= ar.ReadLine();
                                }
                                catch
                                {
                                        Console.WriteLine("Detectado
fin de fichero");
                                        break;
                                }
                                if (contenido==null) break;
                                pos = contenido.IndexOf(" ",10);
                                if (pos==-1) pos=8;

        copia=contenido.Substring(8,pos-8);

                                        total=copia+" ";

                                escribe.Write(total);
                                escribe.AutoFlush=true;

                                total="";

                        }


                }
        }
                //
                // TODO: Add code to start application here
                //
        catch (Exception e)
        {
        Console.WriteLine("Detectado excepcion final"+e);

        }
        ar.Close();
        escribe.Close();
        Console.WriteLine ("FIN");

StreamReader f = new streamReader("C:\\salida2.txt");
StreamWriter fes = new StreamWriter("C:\\salida3.txt");

        string cadena,copiado;
        int posic;
        double entero,entero2;


        cadena=f.ReadLine();
        posic=cadena.IndexOf(" ");
        copiado=cadena.Substring(0,posic);
        entero=Convert.ToDouble(copiado);
```

```
                    while((cadena=(f.ReadLine()))!=null)
                    {
                    posic=cadena.IndexOf(" ");
                    copiado=cadena.Substring(0,posic);
                    entero2=Convert.ToDouble(copiado);
                    double resultado=entero2-entero;
                    string asd=Convert.ToString(resultado);
                    fes.WriteLine(asd);
                    fes.Flush();
                    entero=entero2;

                    }
                    fes.Close();
                    f.Close();

            }

        }

}
```