



***Feasibility study of VoIP in 3GPP UMTS
Release 5 mobile networks interworking
with fixed networks***

By

Petter Andersen & Jon Thoresen

Masters Thesis in
Information and Communication Technology

Agder University College

Grimstad, May 2003



Abstract

The Universal Mobile Telecommunications System (UMTS) is denoted as a 3rd generation cellular system and has been designed with the objective to be a system with global coverage. With improvement of bandwidth capabilities, the UMTS system has the ability to support real time multimedia services. The focus in this thesis is Voice over IP (VoIP) which enables a user to make phone calls in the packet switched network in UMTS.

This thesis starts with a presentation of VoIP with the quality requirements related to a voice session. A voice conversation needs a guaranteed quality to satisfy the participants. This thesis focuses on three main aspects; Quality of Service mechanisms (Best Effort, IntServ and DiffServ), VoIP in UMTS with a certain quality and last but not least implementation of Quality of Service (QoS) in a voice call interworking with external networks.

Best Effort cannot be used when dealing with real time traffic such as VoIP. IntServ reserves resources from the application itself, and gives opportunity for each application in the terminal to request a certain quality. DiffServ works on a higher level and classifies traffic based on type of traffic, not for a particular request. For UMTS interworking with IP networks, the theoretical results suggest that IntServ over DiffServ should be used in the UMTS gateway node.

An evaluation of the UMTS network is done by checking the voice quality attained by the network during a VoIP session in comparison of a traditional circuit switched call setup. Moreover, tests from the Norwegian UMTS network operator NetCom became useful when evaluating how well the VoIP could work when implementing UMTS release 5. The tests were set up with the focus on delay and voice quality in the network, and were meant for disclosing the differences with and without quality parameters during a transmission. Due to network restrictions the test results are limited.



Preface

This thesis is the concluding stage of a Master of Science degree in Information and Communication Technology (ICT) at Agder University College, Faculty of Engineering and Science in Grimstad Norway.

The project has been defined and executed in co-operation with Siemens Germany, Berlin. In this context we would like to thank Reidar Mouhlebe for great supervision and guidance. In addition, he turned out to be a “walking library” with an excellent knowledge of mobile networks.

We would also like to thank Ronny Søbstad at Siemens Norway for his support in helping us to set up the UMTS terminals needed for our test system. Last but not least we would like to thank Daniel Olsson at NordicLan in Sweden for allowing us evaluate the Chariot software needed for these measurements. The cost of a licence would have been financially prohibitive within the framework of this project.

Petter Andersen

Jon Thoresen

Grimstad, Norway 2003



Table of Contents

ABSTRACT	II
PREFACE	III
FIGURE LIST	VI
TABLE LIST	VI
1 INTRODUCTION	1
1.1 THESIS INTRODUCTION	1
1.2 TASK DESCRIPTION	2
1.3 LITERATURE REVIEW	3
1.4 REPORT OUTLINE	4
2 VOICE OVER IP	5
2.1 INTRODUCTION	5
2.2 VOIP SCENARIOS	5
2.3 MOTIVATION FOR INTERNET TELEPHONY	7
2.4 THE FACTORS IN PACKETIZED VOICE	8
2.5 CODING, DECODING AND PROCESSING	9
2.6 VOIP RELATED PROTOCOLS	9
2.6.1 <i>Transport protocols</i>	10
2.6.2 <i>Call processing protocols</i>	10
3 UMTS RELEASE 5	11
3.1 BRIEF NETWORK OVERVIEW WITH GRAPHICS	11
3.2 AIR-INTERFACE, BANDWIDTH AND COVERAGE	13
3.3 SERVICES	15
4 IMPLEMENTING VOIP FUNCTION IN UMTS RELEASE 5	16
4.1 OVERVIEW	16
4.2 IP MULTIMEDIA SUBSYSTEM	16
4.2.1 <i>What is IMS</i>	16
4.2.2 <i>Architecture</i>	17
4.2.3 <i>Elements in the UMTS PS CN</i>	18
4.2.4 <i>IMS elements</i>	18
4.3 VOICE CODECS	23
4.3.1 <i>AMR and AMR-WB alternative codec schemes</i>	23
4.4 PROTOCOLS FOR REAL-TIME TRAFFIC IN UMTS REL.5	24
4.4.1 <i>Transport and control information protocols</i>	24
4.4.2 <i>Signalling protocol</i>	26
5 AN APPROACH TO IP QOS	27
5.1 QoS PARAMETERS AND MANAGEMENT	27
5.2 INTEGRATED SERVICES	28
5.2.1 <i>Integrated Services framework</i>	28
5.2.2 <i>RSVP</i>	29
5.2.3 <i>Service classes</i>	30
5.3 DIFFERENTIATED SERVICES	31
5.3.1 <i>The Differentiated Services Framework</i>	31
5.3.2 <i>DiffServ Domain</i>	32
5.3.3 <i>Differentiated Services Field</i>	32
5.3.4 <i>Per Hop Behaviours (PHBs)</i>	33
6 QUALITY OF SERVICE IN UMTS RELEASE 5	35



6.1	QoS CLASSES IN UMTS	35
6.2	QoS PARAMETERS AND REQUIREMENTS.....	37
6.3	MANAGEMENT OF QoS IN THE UMTS NETWORK	38
6.3.1	<i>Authorization of QoS resources</i>	38
6.3.2	<i>Resource reservation</i>	39
7	UMTS INTERWORKING WITH FIXED NETWORKS.....	41
7.1	OVERVIEW	41
7.2	IP NETWORK INTERWORKING QoS MANAGEMENT	42
8	TESTING THE UMTS NETWORK.....	45
8.1	OVERVIEW	45
8.2	METHOD	45
8.3	VOIP CODECS	46
8.4	EQUIPMENT ARCHITECTURE	47
8.5	SCENARIO 1: SESSION BETWEEN LAPTOP B AND STATIONARY WORKSTATION.	47
8.5.1	<i>Equipment involved in the test</i>	48
8.5.2	<i>Geographical:</i>	48
8.5.3	<i>Setup</i>	48
8.6	SCENARIO 2: SESSION BETWEEN LAPTOP A AND LAPTOP B	49
8.6.1	<i>Equipment involved in the test</i>	49
8.6.2	<i>Geographical</i>	49
8.6.3	<i>Setup</i>	49
9	TEST RESULTS	50
9.1	SCENARIO 1 – TRACE ROUTE	51
9.2	SCENARIO 2 – PART 1, USING THE UPLINK	53
9.2.1	<i>Throughput</i>	54
9.2.2	<i>One Way Delay</i>	55
9.2.3	<i>Jitter</i>	56
9.2.4	<i>MOS Estimate</i>	57
9.3	SCENARIO 2, PART 2, USING THE DOWNLINK	58
9.3.1	<i>Throughput</i>	59
9.3.2	<i>One-Way Delay</i>	60
9.3.3	<i>Jitter</i>	61
9.3.4	<i>MOS Estimate</i>	62
10	DISCUSSION	63
10.1	PRELIMINARY DISCUSSION	63
10.2	VOICE TREATMENT IN UMTS ELEMENTS	64
10.3	QoS	65
10.3.1	<i>Scenario 1</i>	65
10.3.2	<i>Scenario 2</i>	66
10.4	TESTS	67
11	CONCLUSION	68
	REFERENCES	69
	ABBREVIATIONS	71
	APPENDIXES	74



Figure list

FIGURE 2-1: SCENARIOS INTRODUCED WITH PURE IP COMMUNICATION.....	6
FIGURE 2-2: THE VOIP PROTOCOL SUITE [5].....	9
FIGURE 3-1: OVERVIEW OF THE UMTS RELEASE 5 ARCHITECTURE.....	12
FIGURE 3-2: DIVIDING AN AREA INTO DIFFERENT CELLS.....	13
FIGURE 4-1: ARCHITECTURE OF UMTS RELEASE 5 WITH FOCUS ON IMS.....	17
FIGURE 4-2: ARCHITECTURE OF THE CSCF WITH INTERFACES.....	19
FIGURE 4-3: ARCHITECTURE OF MRF.....	21
FIGURE 4-4: USER PLANE PROTOCOL STACK FOR 3G PS CONVERSATIONAL MULTIMEDIA TERMINAL [16].....	24
FIGURE 5-1: INTSERV FRAMEWORK.....	28
FIGURE 5-2: DIFFERENTIATED SERVICES ARCHITECTURE.....	31
FIGURE 5-3: DIFFERENTIATED SERVICES CODE POINTS REDEFINE IP PRECEDENCE IPV4 TOS BYTE.....	32
FIGURE 6-1: AUTHORIZE QoS RESOURCES AT ORIGINATING PDF.....	38
FIGURE 6-2: RESOURCE RESERVATION FLOW WITH SERVICE BASED LOCAL POLICY.....	39
FIGURE 7-1: EXTERNAL NETWORKS INTERWORKING.....	41
FIGURE 7-2: GGSN QoS ARCHITECTURE.....	42
FIGURE 8-1: UMTS TESTING CONFIGURATION.....	47
FIGURE 8-2: SCENARIO 1 MODEL.....	47
FIGURE 8-3: SCENARIO 2 MODEL.....	49
FIGURE 9-1: TRACE ROUTE FROM LAPTOP TO A PC LOCATED AT AGDER UNIVERSITY COLLEGE.....	51
FIGURE 9-2: TRACE ROUTE FROM LAPTOP TO DESTINATION WWW.GOOGLE.COM.....	52
FIGURE 9-3: THROUGHPUT UMTS (UPLINK).....	54
FIGURE 9-4: ONE-WAY DELAY MEASUREMENTS (UPLINK).....	55
FIGURE 9-5: RFC 1889 JITTER MEASUREMENTS (UPLINK).....	56
FIGURE 9-6: MOS ESTIMATE MEASUREMENTS (UPLINK).....	57
FIGURE 9-7: THROUGHPUT MEASUREMENTS (DOWNLINK).....	59
FIGURE 9-8: ONE-WAY DELAY MEASUREMENTS (DOWNLINK).....	60
FIGURE 9-9: RFC 1889 JITTER (DOWNLINK).....	61
FIGURE 9-10: MOS ESTIMATE (DOWNLINK).....	62

Table list

TABLE 6-1: PARAMETERS NEEDED FOR AN END-TO-END VOICE TRANSFERS IN UMTS [11]......	37
TABLE 8-1: MEAN OPINION SCORE ESTIMATES.....	45
TABLE 8-2: CODECS SUPPORTED BY CHARIOT.....	46
TABLE 9-1: TEST SETUP FOR SCENARIO 1, PART 1.....	53
TABLE 9-2: THROUGHPUT RESULTS FROM UMTS (UPLINK).....	54
TABLE 9-3: RESULTS FROM ONE-WAY DELAY MEASUREMENTS (UPLINK).....	55
TABLE 9-4: RESULTS FROM RFC 1889 MEASUREMENTS (UPLINK).....	56
TABLE 9-5: RESULTS FROM MOS ESTIMATE MEASUREMENTS.....	57
TABLE 9-6: TEST SETUP FOR SCENARIO 1, PART 2.....	58
TABLE 9-7: RESULTS FROM THROUGHPUT MEASUREMENTS (DOWNLINK).....	59
TABLE 9-8: RESULTS FROM ONE-WAY DELAY MEASUREMENTS (DOWNLINK).....	60
TABLE 9-9: RESULTS FROM RFC 1889 JITTER MEASUREMENTS (DOWNLINK).....	61
TABLE 9-10: RESULTS FROM MOS ESTIMATE MEASUREMENTS (DOWNLINK).....	62



1 Introduction

1.1 Thesis introduction

Mobile communications is one of the fastest growing businesses in the telecommunications market. Service and equipment costs are becoming increasingly affordable and users are starting to get used to the ease of connectivity. With the development of the wireless packet service in the 3rd generation cellular networks, it will be possible to make use of this to extend the IP traffic considerably.

A new feature introduced in the 3rd generation mobile system UMTS release 5 will be VoIP [1]. This means that the UMTS operators will, in addition to data communication, use the packet domain for voice calls to reduce costs in their networks [2]. This differs from traditional telephony where a definite amount of resources are dedicated for each call; a VoIP session can be implemented with different coding techniques making it possible to adjust the need for bandwidth resources. In addition to bandwidth adjustments the UMTS network need QoS management and suitable protocols.

A voice packet call is a real time application in need of guaranteed service, making “best effort” insufficient when the network is congested or suffers from overload. It is important to do research on which mechanism is available and which is ideal for QoS in mobile networks, to preserve the end user’s need for voice quality and minimum transmission delay.

UMTS interworking with external IP networks introduce a challenge for QoS management between mobile subscribers and Internet participants. This leads to a co-operation between UMTS gateways and IP networks concerning packet handling. The public Internet is an IP network with delivery by “best effort” as default.

The Integrated Services (IntServ) approach provides services end-to-end, from an application at the host, to the other party’s application. IntServ requires a large amount of signalling between nodes, and reservation information is stored in each node in the network for individual flows.

The Differentiated Services (DiffServ) approach provides service using simple mechanisms to divide different traffic classes on behalf of the priority field in the IP header. DiffServ requires neither state information per flow nor signalling to each node, which increases scalability in large networks.

To determine which mechanism is suitable for UMTS with regard to VoIP, a broad evaluation of the network and the set-up procedures are required. In addition, we need to identify the possibilities for implementing the QoS mechanisms in a network such as UMTS.



1.2 Task description

The title for this thesis is “Feasibility study of VoIP in 3GPP UMTS Release 5 mobile networks interworking with fixed networks”.

The UMTS is the beginning of a new mobile communication system, which is classified as the third generation mobile system. UMTS has greater bandwidth than earlier mobile systems such as GSM/GPRS, and can then easily offer mobile multimedia applications such as video conferencing, location based services and image downloads. The UMTS roll-out is performed in steps and will expand extensively in the years to come. “Release 99” is the first commercial network that is brought to life.

3rd Generation Partnership Project (3GPP) UMTS release 5 specifies VoIP in relation to multimedia services. Multimedia services are being realized with a new domain in the UMTS Core Network, IP Multimedia Subsystem (IMS). The IMS will assure multimedia traffic from a user terminal to the Public Land Mobile Network (PLMN), Public Switched Telephone Network (PSTN) and IP networks such as the public Internet. Even though both PSTN and Internet are defined as fixed networks our main concern is IP end-to-end, excluding the focus on PSTN networks. To achieve a quality voice call in such networks, there have to be mechanisms securing QoS.

To find out how to implement QoS in a mobile network, we will have to make an evaluation of the QoS parameters needed to support a packet voice session. Further on, research is needed to identify the parameters and mechanisms that are realistic for the IMS in regard to exchange of information between the IMS and a user terminal.

Tests in the UMTS network will give feedback on how well the network handles packet voice calls according to different quality classes and codecs. Codecs is used for the purpose of controlling the transmitted load in the network. The outcome of tests will back up our feasibility study of VoIP calls in a mobile network with regards to an end-to-end session.

The standardization from 3GPP for UMTS release 5 is not yet completed and there is no IMS system for mobile VoIP existing today. It is of great interest to Siemens to have this feasibility study done on the probable success of such a system, and to gain knowledge about VoIP in fixed and mobile networks. The idea for this thesis was established when the students wanted to work with data-communications and had a great interest for mobile networks. Siemens on the other hand, has to implement QoS in its mobile networks specified by 3GPP and need knowledge about QoS in regards to VoIP.



1.3 Literature review

This section explains where we have obtained relevant information for this thesis. We have used information from various sources which will be further described.

The most interesting site must be 3GPP [3]. The 3GPP is a large open international community consisting of well known organizational partners such as the European Telecommunication Standard Institute (ETSI). The Technical Specifications (TSs) from 3GPP are Siemens main sources when developing the IMS, and therefore also our main literature source as well.

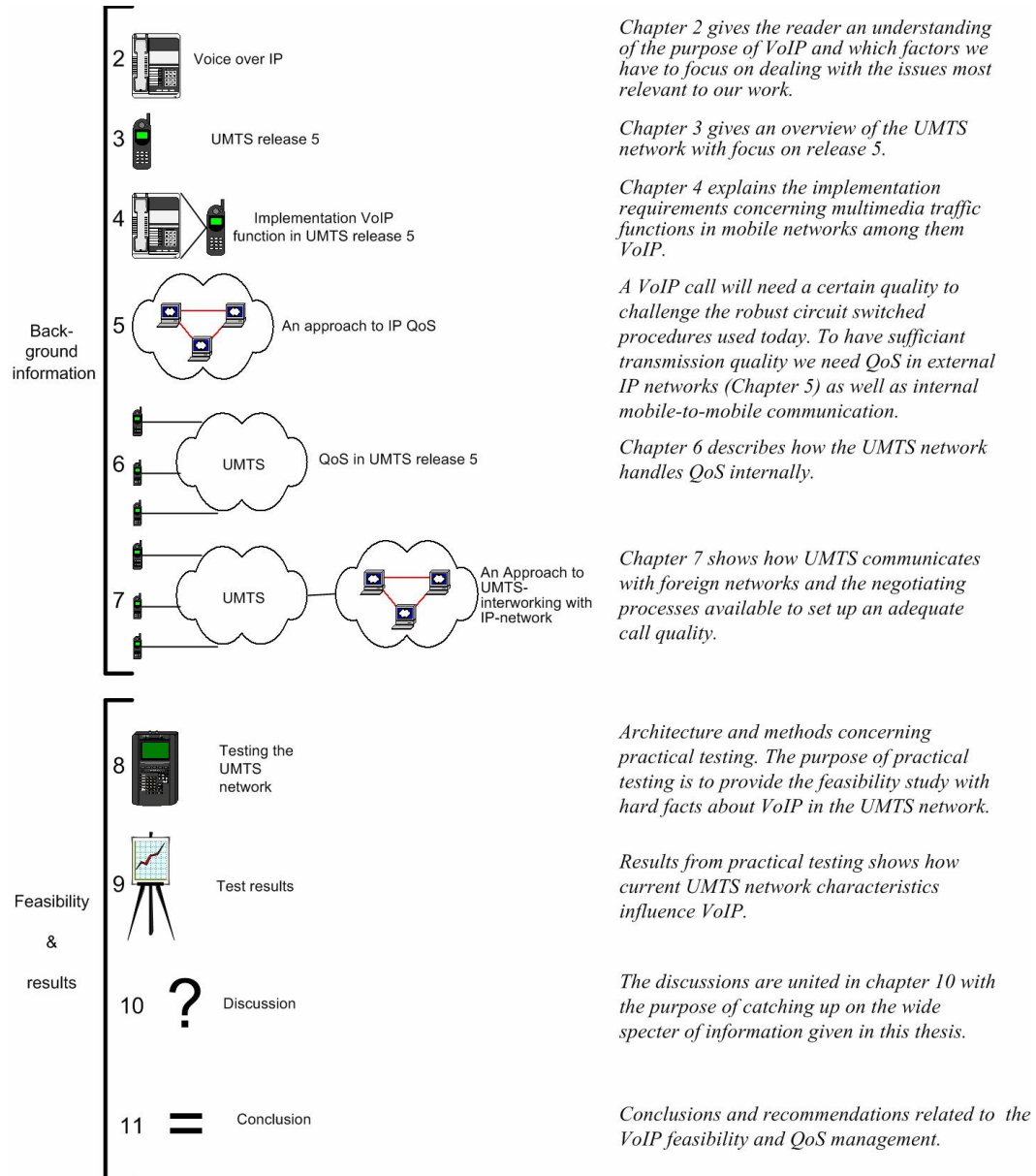
Internet Engineering Task Force (IETF) [4] has an open site for new Internet standards collected in a database. The specifications are given as Request For Comments (RFCs) and drafts to for new Internet standards, written by researchers concerned with the evaluation of the Internet architecture and its smooth operations.

3GPP and IETF respectively produce specifications meant for highly competent experts. To support our understanding we have read reproductions from other sources to gain an overview expertise. White-papers and Internet drafts in particular have turned out to be useful reading. Since few reliable sources exist on the Internet we have used the reproductions as complementary information.



1.4 Report Outline

This report is structured in ten chapters which will be described briefly. In this description we have made the left-hand symbolic features play a role for the total understanding.



Due to complexity and the large amount of technical information, we have decided to sort our references the same way RFC references are sorted; with numbering. The first reference used in this report is numbered “1” and can be found in the reference list.



2 Voice over IP

Interactive multimedia traffic is increasing as applications like streaming media and packet telephony grow in popularity. It is necessary to investigate the characteristics of this traffic, particularly because its behaviour in the face of network congestion differs from that of the currently dominant TCP traffic. VoIP as part of the interactive multimedia traffic is our main concern. It is categorized as a real-time service because the voice packets cannot tolerate long delays in delivery.

This chapter describes the varieties of the term “Voice over IP” and which scenarios focused on in this thesis. We will investigate the reasons why the communication industry is so keen on VoIP. This will be followed by a study of the parameters and the difficulties and concerns due to VoIP as real time traffic in a network. Finally, we will describe the importance of codec technology and VoIP related protocols.

2.1 Introduction

VoIP is when a voice is digitized and transported via packet networks managed by the Internet Protocol (IP). It is a service that requires real-time transfer over an end-to-end session. Several terms have been used to describe VoIP, thus it is necessary to clarify some terms:

- Internet telephony: Telephone calls over the public Internet. This term implies the use of IP telephony and packet-voice.
- internet telephony: Telephone calls in a private network (an internet). This term also implies the use of IP telephony and packet-voice.
- IP telephony: The use of IP to forward voice calls through the Internet or an internet. This term implies the use of packet-voice.
- Packet voice: The use of packet switched networks instead of the telephone circuit-switched network to transport telephone calls.

2.2 VoIP scenarios

UMTS terminals will communicate with IP after signalling with the IMS and new scenarios are introduced in this way. New paths are opened for a user if the network operator prepares it. Today we can make phone calls with IP e.g. with MSN Messenger on a PC connected to Internet. A session where we use packet voice to an operator, then routed to the other party on the PSTN-network with traditional circuit switched TDM (Time Division Multiplexing) frames. With UMTS rel.5 we could imagine that an adequate IP call can be arranged if the quality is good enough. To illustrate where a call can be made we have designed Figure 2-1.

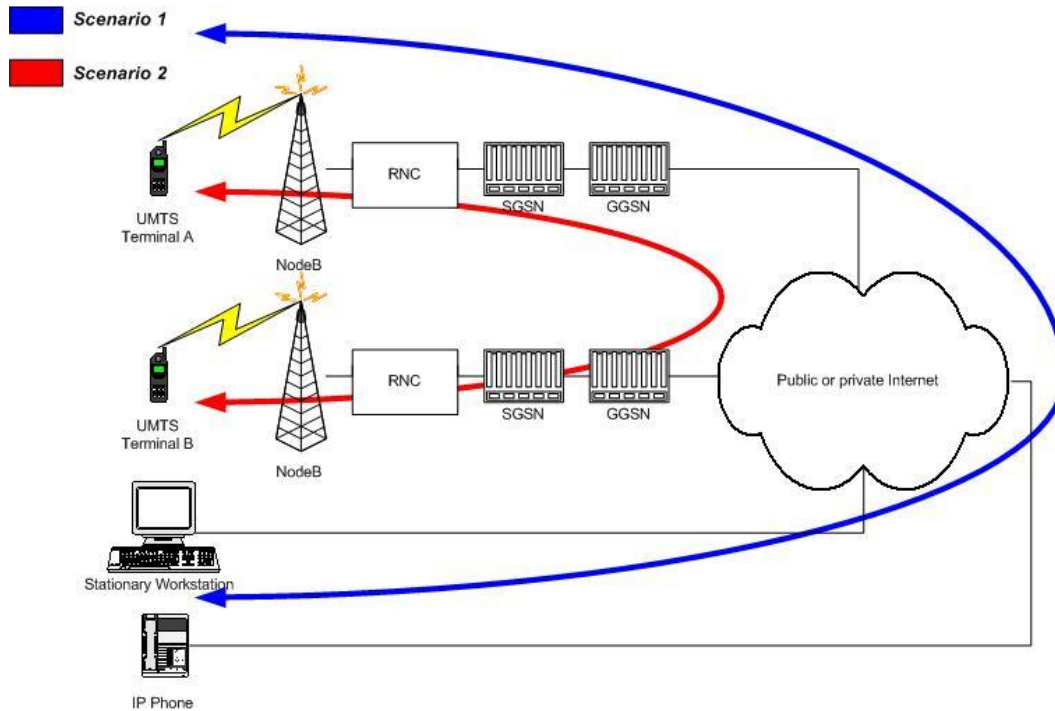


Figure 2-1: Scenarios introduced with pure IP communication

The scenarios including VoIP in UMTS Release 5 are:

Scenario 1: VoIP call from UMTS terminal to a Stationary Workstation or an IP phone on the Internet.

If a user want to make a VoIP call from a UMTS terminal to a PC or IP phone located somewhere in the Internet, the quality need to be arrange between the PCs belonging network or the IP phones gateway and the GGSN.

Scenario 2: VoIP call from an UMTS terminal to another UMTS terminal.

If we want to make an IP phone call from a UMTS terminal to a UMTS terminal belonging to foreign network, the quality need to be arrange between two GGSN nodes.

“Scenario 3”: VoIP call from UMTS terminal to a telephone on the PSTN

This scenario is not in our concern because the arrangement of the quality is the same as in the first scenario. The quality of service is just a concern from the UMTS terminal to the gateway that terminates the transformation from the UMTS traffic to PSTN traffic. The traffic in the PSTN network does not suffer from bad quality.

The two first scenarios will further be discussed in this thesis; these are the main aspects to solve, defined in our thesis description.



2.3 Motivation for Internet telephony

IP telephony is viewed by some people to be an attractive and effective technology and by others as nothing more than irritant. The reason why some think it is irritating is because of the speech quality and the overall ability of the Internet to support voice traffic. There are four major reasons why the communication industry is so keen on VoIP [5].

1) The business case

There is a compelling business case for the deployment of the IP protocol suite and associate equipment to support telephony services. We can summarize this by way of the following suppositions.

It is clear that *integration* of voice, data and video traffic will be demand by multi-application software. Web servers can then interact with the customers with real-time data, voice and video images.

The integration of voice and data allows for *bandwidth consolidation*, which fills up the communication channel more efficiently than conventional voice link. In a TDM based network, the user is given bandwidth even when the user is not talking; this wastes about 50 % of the channel. A packet-based speech channel only sends when there is something to send. Modern analog-to-digital operation in a packet-based speech channel can operate at about 4.8 to 8 kbps, in contrast to TDM telephony channels that operates at 64 kbps. The bandwidth ratio is just over 1:10 in favour of the packet-based network.

The telephone *infrastructure* is based on technology that was developed decades ago. Almost everything was hardwired and not easy to change. Even though low bandwidth codecs operating in the 5 to 8 kbps range have been in the industry for many years, the telephone switches and other telephony components could not take advantage of this because they could not handle other than 64 kbps TDM. One of the key aspects of VoIP rests on the idea of an infrastructure that supports change and on the ability to negotiate services, as well as a different level of technology. By using the Internet backbone to make a phone call, lead to passing of the toll service of public switched telephone network. This approach avoids costly charges in the telephone network.

The above indicates that it could be economically profitable to move towards telephony IP

2) Universal presence of IP

The second major reason for IP telephony is the universal presence of IP in associated protocols in user and network equipment. The key importance is that IP resides in the end-use workstation in contrast to competitive technologies such as ATM (Asynchronous Transfer Mode) and Frame relay that operates as UNI (User Network Interface). The existence of IP in personal computers, mobile equipment and workstations gives IP a decided advantage over other existing technologies.



3) Maturation of Technology

The maturation of technologies makes VoIP feasible. The DSPs (Digital Signal Processors) that are found in codecs and high-speed modems are supporting VoIP on a wide scale. Their performances have opened the way for support of applications that were unthinkable just a few years ago. The end users are not satisfied with applications that just display text-only images. Maturation of three key technologies will foster a revolution in information technology. They are: increasing bandwidth of communication links, increase capacity of CPUs, and the advent of a reusable, intelligent plug-and-play software code. These maturing technologies will lay the groundwork for a new generation of applications.

4) The shift to data networks

The world is experiencing a shift away from circuit-based networks to packet-based network. There are two reasons for this change. First, the data industry is more a growth than the voice industry. Second, the bandwidth and QoS requirements for a voice conversation do not change. The processors in computer and mobile equipment become faster as users demand better and better services.

2.4 The factors in packetized voice

There are four key factors to evaluate when placing speech traffic on public or mobile networks. The key factors are: packet time, bandwidth requirements, packet loss and computational effort. The reason to evaluate these factors is to fulfil the demands for a real-time transfer and a qualitative speech.

The first factor is *packet time* and it deals with two performance operations, constant and variable delay: Constant delay, or latency, is the time required for transmission of data packets from origin to destination. The delay occurs from bandwidth sharing, processing and queuing at routers and endpoints. It causes two problems: echo and talker overlap. Echo is caused by the signal reflections of the speakers' voice from the far-end telephone equipment into the speakers' ear. Talker overlap arises when one talker is stepping on the other talker's speech. Variable delay, or jitter, is the result of packets arriving at irregular intervals. Severe jitter in IP voice transmissions causes jittery or shaky voice quality. Removing jitter is done by allow the slowest packets to arrive in time to be played in correct sequence.

The second factor deals with how much *bandwidth* is required to support the transmission. The bandwidth must take care of transmitting speech and overhead that support the signals.

The third factor, *packet loss*, is a result of voice traffic that is congested by the network because of large queues in routers. The packets are discarded and result loss of speech phonemes.



The fourth factor is the *computational effort* need to support the coding, transport, and decoding of speech image in each machine. Computational effort refers to the expense and complexity involved in supporting the audio applications. Processors described in next chapter can do the effort.

Some of these factors have their own parameter in a set of quality services discussed in Chapter 5.1.

2.5 Coding, decoding and processing

To be able to send voice effectively on the Internet it is important to use a Codec. The term denotes a piece of software or hardware, which encodes and decodes signals. The 'voice Codec' in a mobile phone is responsible for converting voice into bit strings and converting bit strings received back into voice. To do this, it uses sophisticated data reduction and data compression techniques. The specific codecs used with UMTS VoIP calls will be discussed in Chapter 4.3.

Digital signal Processors (DSPs) are the engine for voice codecs. It is a specialised processor that has been used for many years in telephony application such as mobile/wireless networks. They are attractive because they are small, fast and use little power.

Together the codecs and DSPs are responsible for the amount of bandwidth a packet based voice call need.

2.6 VoIP related protocols

There are several protocols that support voice traffic on the Internet. Here we describe some of the protocol structures.

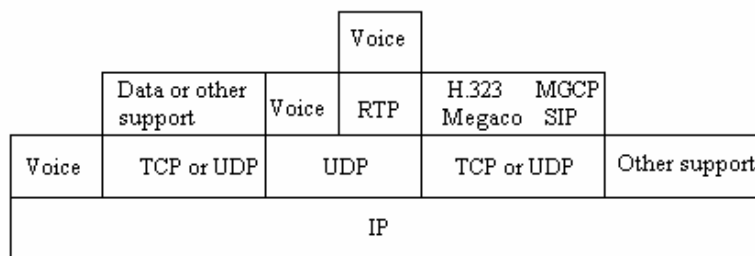


Figure 2-2: The VoIP protocol suite [5]

Figure 2-2 shows several of the Internet protocols and their relation to the conventional Internet layer model. One interesting aspect is the placement of the voice operations in three possible positions. Voice can run directly over IP; or over the UDP (User Datagram Protocol) then IP; or over the RTP (Real-time Traffic Protocol), then UDP, and then IP.



2.6.1 Transport protocols

The most commonly used procedure for voice operations in IP networks is RTP/UDP/IP. The RTP is designed for supporting end-to-end real-time traffic. This traffic requires play-back at the receiver in a time-sensitive mode, as for voice and video systems. The RTP also operate with both unicast and multicast applications. It provides services that include payload type identification (type of codec etc.), sequence numbering, timestamp and delivery monitoring. The use of UDP under RTP is to make use of the multiplexing and checksum service. The sequence numbers in RTP allows the receiver to reconstruct the sequence of the packets, and be used to determine the proper location of a packet. The use of all these protocols to transfer a voice on a network leads to a lot of header information in the packets. The header information has no value for a user of the system, and it will increase the need of bandwidth.

Note that RTP itself does not provide any mechanism to ensure timely delivery, address resource reservation or other quality-of-service guarantees. It does not guarantee delivery or prevent out-of-order delivery, nor does it assume that the underlying network is reliable and delivers packets in sequence [6].

2.6.2 Call processing protocols

Several protocols in Figure 2-2 provide the control and management of telephony sessions in an internet. They are known as signalling or call processing protocols, and their job is to set up clear voice or video calls in a packet network. The protocols can run over TCP or UDP. As an examples on one of these protocols we can mention Session Initiation Protocol (SIP) [7], this will further be described in Chapter 4.4.2

When we now have gained knowledge concerning the VoIP, it is important to get some information about the network were the VoIP is going to be transferred. The next chapter will give information of the new UMTS Release 5 network.



3 UMTS Release 5

This section presents a brief introduction to 3GPP UMTS Release 5. We will attempt to give the necessary background for further reading. In this thesis we will focus on the packet switched domain and give a presentation of the architecture, bandwidth, coverage, and services in UMTS.

The 3GPP is a global standardization initiative created in December 1998. The goal of 3rd Generation systems is to give mobile users high-speed access to a wide range of multimedia services, wherever they are in the world. It is thus a globally standardized technology, and is often referred to as UMTS. It has been designed to permit global roaming with the widest choice of user terminals, and to interoperate with the other 3rd Generation technologies, as well as with 2nd Generation systems such as GSM. It is compatible with, and will facilitate the evolution of existing GSM infrastructures [8].

3.1 *Brief network overview with graphics*

Release 99 was only the start of the UMTS releases. Release 4 was frozen in March 2001, providing more features and enhancements including preliminary definitions of “IP Multimedia” networks. Release 5 represent a major shift in the UMTS architecture, it was scheduled for early 2002. Release 5 proposes to provide full IP Multimedia capability and to offer both traditional telephony as well as packet switched services over a single converged packet based network. Some of the motives for changing to full IP Multimedia capability are mention in Chapter 2.3.

The architecture of UMTS release 5 is designed to add new features like:

- Both IPv4 and IPv6.
- All-IP Multimedia Subsystem
- All-IP UTRAN (UMTS Terrestrial Radio Access Network)
- High speed downlink packet access
- Better Quality of services
- Full separation of transport and control commenced in Rel.4

These features are designed to meet the trends that can be recognised in the telecommunications market today. The users have an increasing need to gain access to information in as efficient manner as possible. At the same time they have an increasing need to be mobile [2].



When we now present the architecture of the UMTS Release 5, the focus has been on the elements in the network concerning VoIP. That is network element for circuit switched traffic is not present, see Figure 3-1.

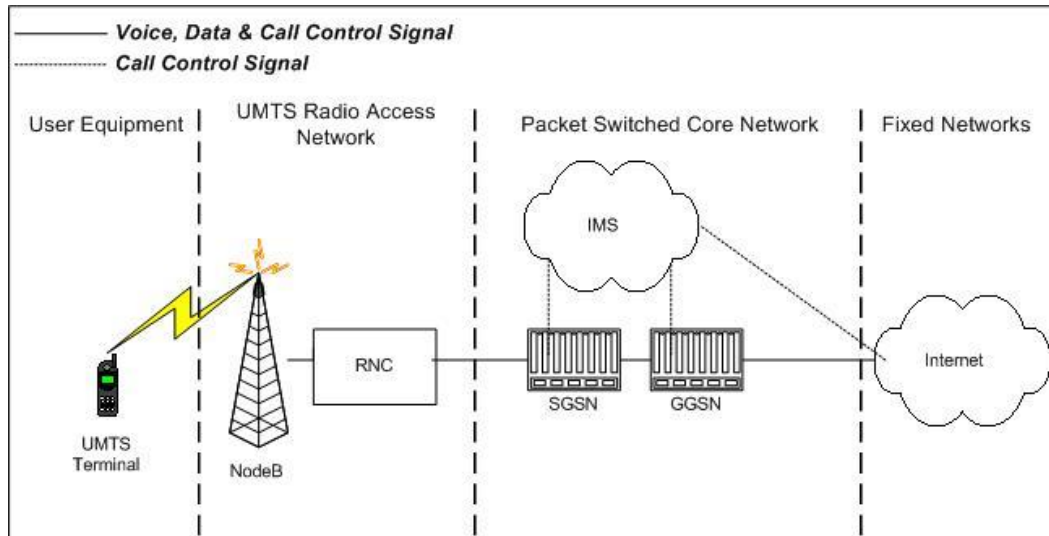


Figure 3-1: Overview of the UMTS Release 5 architecture

The main UMTS Release 5 network domains concerning VoIP are: User Equipment (UE), UMTS Terrestrial Radio Access Network (UTRAN) and Packet Switched Core Network (PS-CN) with IMS (IP Multimedia Subsystem) [9].

- The UE is in our case a cellular phone that communicates with UTRAN via the air-interface.
- UTRAN provides the air interface access method, radio mobility and resource utilisation for the UE. The coverage provided by UTRAN is described in next chapter.
- UMTS PS-CN provides switching, routing, and transit for user traffic to and from the Internet, PSTN and other mobile users. It also stores databases and provides network management functions. Packet switched elements are Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN) [10]. These nodes are further described in Chapter 4.2.3.
- The IMS connects to Internet as well as PSTN/ISDN. It may terminate voice and multi-media calls on Internet and PSTN/ISDN. This lead to further study of the IMS. Due to complexity and the large amount of information, the IMS will further be discussed in Chapter 4.2.



3.2 Air-Interface, Bandwidth and coverage

UMTS use the innovative radio interface known as UTRA, the UMTS Terrestrial Radio Access. This interface is using Wideband Code Division Multiple Access (W-CDMA) technology. W-CDMA uses a set of spreading sequences or codes with optimal correlation characteristics [11].

Each wireless radio network is geographically partitioned into cells. One ground station forms a cell, and supplies it with radio signals. Each cell has a certain capacity of communication channels. The smaller the cells are planned, the more cells must be developed and consequently more ground stations with antennas and processing electronics. The high frequencies used in UMTS, 2000MHz, will result in rather small cells, because of bad spreading of such frequencies. Substantial difference to GSM is that with UMTS all adjoining cells use same frequency channels for communication. Since UMTS uses the W-CDMA multiplexing, it does not come to a mutual influence of neighbouring cells.

A substantial characteristic of W-CDMA is that the maximum data rate possible depends on the transmitting power and thus also on the distance between UE and ground station (Node B). The larger the cell is, the smaller the possible data rate becomes at the edge of the cell. The UMTS offer smaller data rates for a moving participant through the cell and when the population decreases.

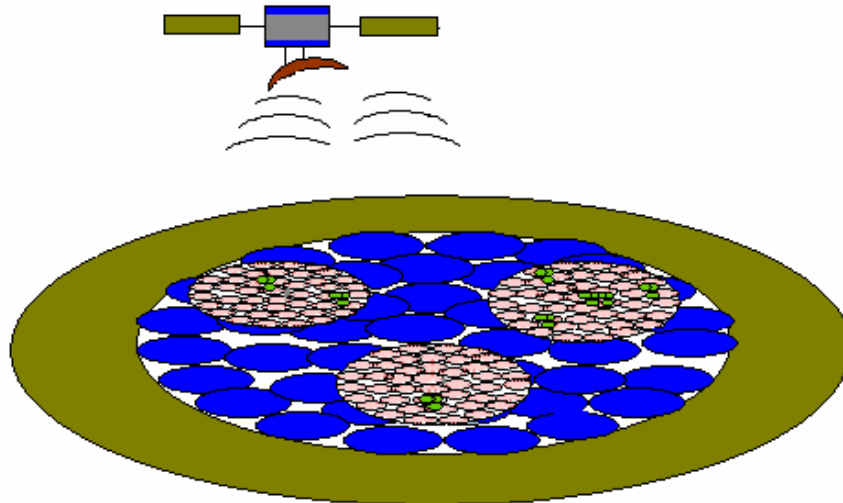


Figure 3-2: Dividing an area into different cells

The Figure 3-2 shows the cell hierarchy. The large beige cell is the cell supplied by satellites, called world cells. The entire national territory is represented as an ellipse completely filled up with blue macro cells. The three population centres are filled up in each case by pink micro cells. Hotspots are represented as small green points, which are positioned over the micro cell level. These are required for very large data rates.



- **World cells**

The World cells are realized in the future by satellite binding, whereby radio services can be offered at geographically inconvenient places like the desert or the ocean. The World cells are realized by its own radio interface, which does not have to do anything with the UTRAN. Type of technologies used for the satellite network is not yet decided.

- **Macro cells**

The macro cells are dimensioned quite spaciouly. The largest cells have for instance an expansion of for instance 2 km. In macro cells the largest surface covering data rate is limited to 144 kbps. If however a user is in the proximity of the ground station, it can be assigned higher data rates, up to 2 Mbps. A mobile participant, who is supplied by macro cells, can move with a maximum speed of 500 kmph. Macro cells have to supply a country with all the UMTS services, whose data speed are however reduced. Since the cells are relatively large, fewer antenna towers have to be set up, which helps to save costs.

- **Micro cells**

The micro cells are already smaller dimensioned than the macro cells and have an expansion of for instance 1 km. The cells are capable of providing a surface covering maximum data rate of 384 kbps. This surface covering data rate corresponds to six times the ISDN speed. The small dimensioned cells result in more cells, and will lead to higher costs. For this reason one will use micro cells mainly in areas with larger population density such as regions around cities. Micro cells offer a higher surface covering maximum data rate than the macro cells, but the participant cannot move faster than 120 kmph.

- **Pico cell**

The pico cells have an expansion of only 60 m and permit a surface covering data rate up to 2 Mbps. These mini cells are realized only at places where such high transmission rates are needed. The Pico cells takes primarily place within the indoor range (in buildings) and are relevant for business areas, airports, stations, conference centres, stock exchanges etc. The maximum speed of the participants is limited to 10 kmph in pico cells, which should not incur any problems in office buildings [11].



3.3 Services

The main reason why the UMTS Release 5 is developed is to give the opportunity to establish new services [11]. The new services are a result of the new architecture, transmissions techniques and nodes. GSM has low bit rate and limited services.

Types of multimedia services in Release 5:

- Interactive conversational services
 - Real-time 2 (or n-) way human speech, very low latency
 - Multimedia call/conferences (even more QoS/BW req.)
 - Voice telephony is one example here
- Interactive games
 - May allow more latency in some cases
- Streaming video-clips
 - Allowing some delay
- Downloading music, films, files, logos, and ring tones
- Chat, SMS, email



4 Implementing VoIP function in UMTS release 5

UMTS release 5 is being designed for handling VoIP sessions. The main features and equipment to support such a session are described as follows:

4.1 Overview

In this chapter we will turn our attention to give an understanding on how the UMTS release 5 can handle a real-time session such as VoIP. The main topics that provide building blocks for a successful VoIP implementation are nodes to handle processing, signalling and bearer traffic, codecs and IP telephony-related protocols. In addition to these requirements we have to implement QoS. QoS is the main task in this thesis and will be discussed separately in Chapter 5 and 6.

4.2 IP Multimedia Subsystem

IMS (IP Multimedia subsystem) [12] are a set of new nodes in the UMTS Release 5, these elements are necessary to support IP multimedia services in the UMTS. This section will give an overview on the nodes in IMS standardized by the 3GPP supporting VoIP sessions. It will help us understand how to deal with the scenarios described in Chapter 2.2. IMS Solutions from Siemens thorough described in Appendix A.

4.2.1 What is IMS

Since the IMS is under development there are several version of the system available today, this result in different solutions from companies like Siemens, Lucent and other. IMS is the solution made for enabling a vision of robust, high-value services that integrate multimedia activities and allows services to interact with each other. This will enhance the natural process of the end user whether they are operating in fixed, mobile or roaming networks. The integration and interaction of media types opens new paths for richer services than those available in systems today. The essential aspect of IMS is an IP based transport mechanism for both real-time and non-real-time services.

The IMS enables person-to-person IP connections between terminals, which will give rise to many new multimedia services like interactive gaming, "see what I see video telephony" services, and end-to-end simultaneous voice, video and chat sessions between terminals.

In particular, IMS moves real-time services, from the Circuit Switched (CS) Domain to the Packet Switched (PS) Domain, supported by Internet Protocols (IP) rather than the GSM MAP protocols, which have their origins in ISDN. It means that all services will eventually be delivered via one integrated network rather than two overlaid networks.



4.2.2 Architecture

As we have seen in Chapter 3.1, the IMS utilizes the PS domain to transport multimedia signalling and bearer traffic. The IP multimedia subsystem is independent of the CS domain although some network elements may be common with the CS domain. To realize IMS as specified by 3GPP in UMTS Release 5, several logic nodes are needed.

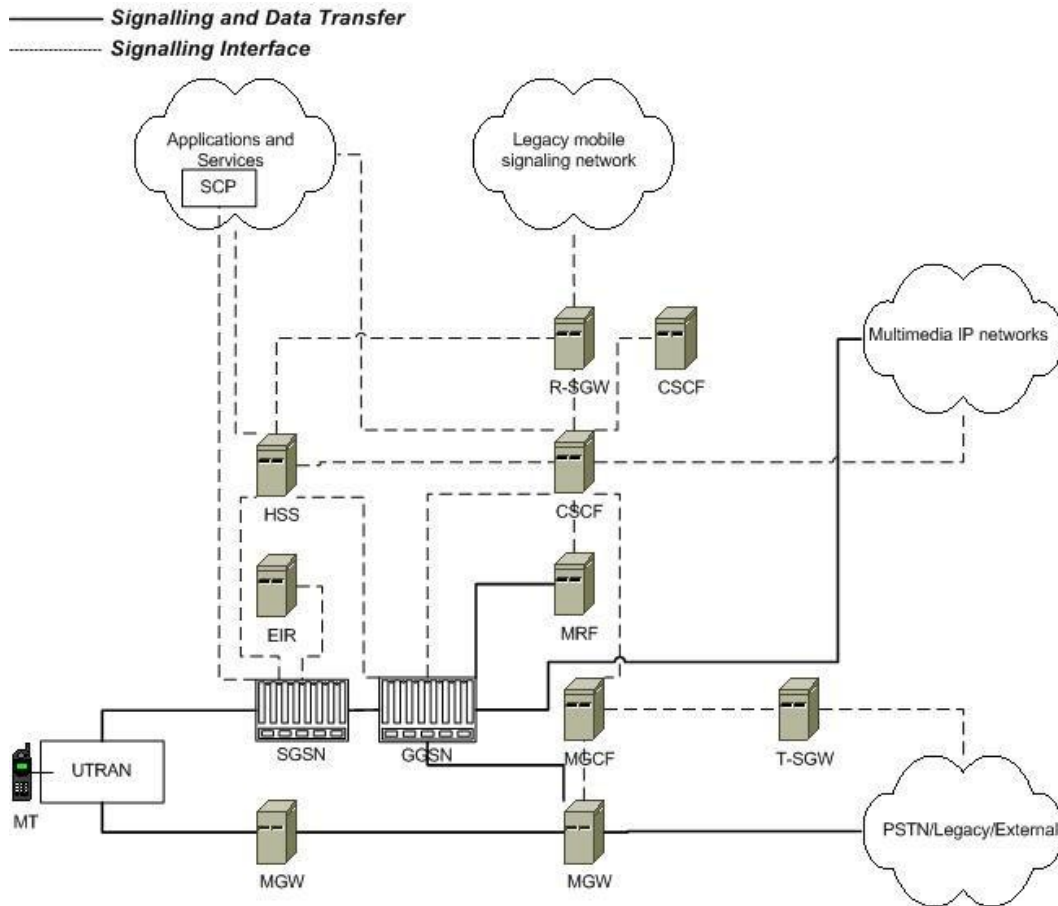


Figure 4-1: Architecture of UMTS Release 5 with focus on IMS

The elements shown in Figure 4-1 present the IMS domain and the packet switched core network domain for UMTS, with regards on further explanation on how to do a VoIP session. The elements in this figure are presented in greater detail.



4.2.3 Elements in the UMTS PS CN

As described in 3.1 the UMTS CN consists of GGSN and SGSN [13]. They constitute the interface between the radio system and the fixed networks for packet switched services and performs all necessary functions in order to handle the packet transmission to and from the mobile stations.

Serving GPRS Support Node (SGSN)

A SGSN is responsible for the delivery of data packets from and to the mobile stations within its geographical service area. Its tasks include packet routing and transfer, mobility management (attach/detach and location management), logical link management, and authentication and charging functions.

Gateway GPRS Support Node (GGSN)

A GGSN acts as an interface between the UMTS backbone network and the external packet data networks (radio network and the IP network). It converts the packets coming from the SGSN into the appropriate format and sends them out on the corresponding packet data network. The GGSN also performs authentication and charging functions. The location-register function in the GGSN stores subscriber data received from the Home subscriber Server (HSS) and the SGSN. HSS is further described in next chapter

4.2.4 IMS elements

The IMS consists of elements described her, among some circuit switched elements not mention.

Home Subscriber Server (HSS)

This is the master database for all users. This includes the HLR (Home Location Register) in the former GPRS network. It is responsible for keeping a master list of features and services associated with a user. The list is used for tracking of location of and means of access for its users. It provides user profile information, either directly or via servers. These subscription profiles hold: User identities, subscribed services and profiles, service specific information, mobility management information and authorization information. The HSS consists of IP multimedia functionality to transfer information to the appropriate entity in the CN to provide call/session establishment, security, authorisation, identification, etc. It also access authentication servers (e.g. AUC, AAA).



Call State Control Function (CSCF)

The CSCF can act as Proxy CSCF (P-CSCF), Serving CSCF (S-CSCF) or Interrogating CSCF (I-CSCF).

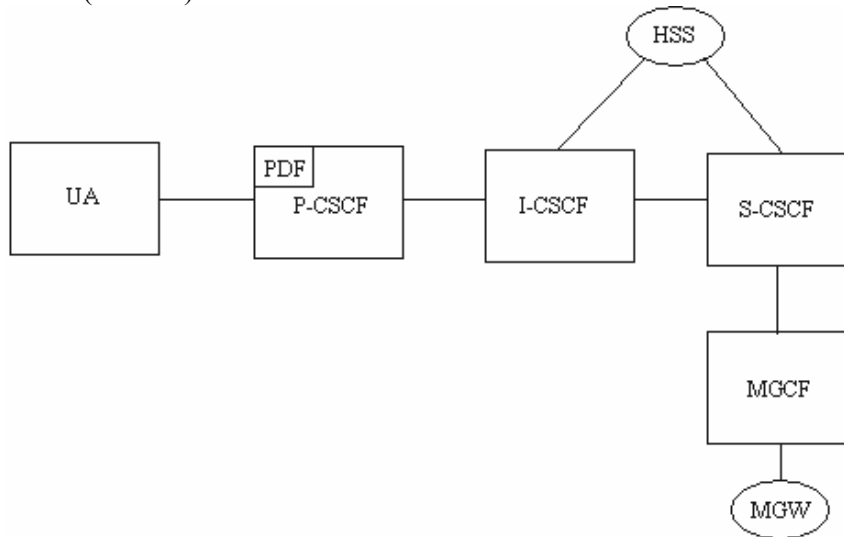


Figure 4-2: Architecture of the CSCF with interfaces

Proxy CSCF (P-CSCF)

The Proxy-CSCF (P-CSCF) is the first contact point for the UE within the IM CN subsystem. The P-CSCF behaves like a Proxy, i.e. it accepts requests and services them internally or forwards them on. The P-CSCF may behave as a User Agent, i.e. in abnormal conditions it may terminate and independently generate SIP transactions. The Policy Decision Function (PDF) is a logical entity of the P-CSCF. The functions performed by the P-CSCF are:

- Forward the SIP messages and requests to and from UE.
- Maintain a Security Association between itself and each UE.
- Should perform SIP message compression/decompression.
- Authorisation of bearer resources and QoS management.



Interrogating CSCF (I-CSCF)

Interrogating-CSCF is the contact point within an operator's network for all connections destined to a user of that network operator, or a roaming user currently located within that network operator's service area. There may be multiple I-CSCFs within an operator's network. The functions performed by the I-CSCF are:

- Registration
 - Assigning a S-CSCF to a user performing SIP registration
- Session-related and session-unrelated flows
 - Route a SIP request received from another network towards the S-CSCF.
 - Obtain the Address of the S-CSCF from HSS.
 - Forward the SIP request or response to the S-CSCF determined by the step above

Serving CSCF (S-CSCF)

Serving-CSCF performs the session control services for the UE. It maintains session state needed by the operator for support services. Within an operator's network, different S-CSCFs may have different functionalities. The functions performed by the S-CSCF during a session are:

- Accepts Register requests and makes the information available through the location server (e.g. HSS).
- Session flows:
 - Session control for the registered endpoint's sessions
 - Interaction with Services Platforms for the support of Services
 - May terminate and independently generate SIP transactions
 - Provide UE with service event related information (e.g. notification of tones)
 - Forward the SIP request or response to the I-CSCF or P-CSCF on behalf of a subscriber



Media Gateway Control Function (MGCF)

This component is PSTN/PLMN termination point for a defined network.

The MGCF:

- Controls the parts of the call state that pertain to connection control for media channels in a MGW, described later.
- Communicates with CSCF.
- MGCF selects the CSCF depending on the routing number for incoming calls from legacy networks.
- Performs protocol conversion between the Legacy (e.g. ISUP, R1/R2 etc.) and the R00 network call control protocols.
- Out of band information assumed to be received in MGCF and may be forwarded to CSCF/MGW.

Multimedia Resource Function (MRF)

The architecture concerning the Multimedia Resource Function is presented in the figure that follows.

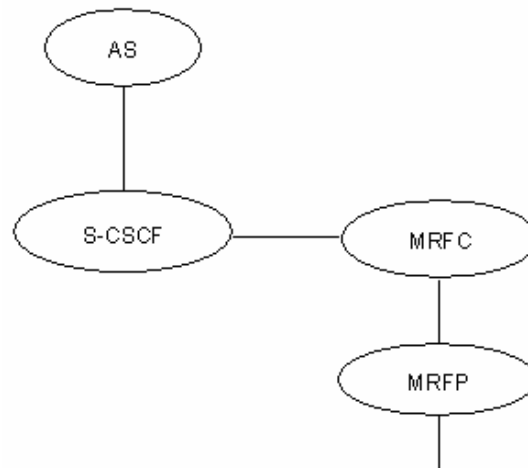


Figure 4-3: Architecture of MRF

The MRF is split into Multimedia Resource Function Controller (MRFC) and Multimedia Resource Function Processor (MRFP) as shown in Figure 4-3.

Tasks of the MRFC are the following:

- Control the media stream resources in the MRFP.
- Interpret information coming from an AS and S-CSCF (e.g. session identifier) and control MRFP accordingly.



Tasks of the MRFP are the following:

- Control of the bearer Between MRFP and GGSN
- Provide resources to be controlled by the MRFC.
- Mixing of incoming media streams (e.g. for multiple parties).
- Media stream source (for multimedia announcements).
- Media stream processing (e.g. audio transcoding, media analysis).

IP Multimedia Subsystem - Media Gateway Function (IMS-MGW)

An IMS-MGW may terminate bearer channels from a switched circuit network and media streams from a packet network (e.g. RTP streams in an IP network). The IMS-MGW may support media conversion, bearer control and payload processing (e.g. codec, echo canceller, conference bridge), it:

- Interacts with the MGCF for resource control.
- Owns and handles resources such as echo cancellers etc.
- May need to have codecs.

The IMS-MGW will be provisioned with the necessary resources for supporting UMTS/GSM transport media. Further tailoring may be required to support additional codecs and framing protocols, and support of mobile specific functions.

Signalling gateway function (SGW)

A Signalling gateway function (SGW) is used to interconnect different signalling networks i.e. SCTP/IP based signalling networks and SS7 signalling networks. The signalling gateway function may be implemented as a standalone entity or inside another entity. The session flows in this specification do not show the SGW, but when interworking with PSTN/CS domain, it is assumed that there is a SGW for signalling transport conversion. The SGW is implemented as two logical nodes.

Roaming Signalling Gateway (R-SGW)

The role of the R-SGW related only to roaming to/from 2G/R99 CS and GPRS domain to/from R00 UMTS Teleservices domain and UMTS GPRS domain and is not involving the Multimedia domain. In order to ensure proper roaming, the R-SGW performs the signalling conversion at transport level



Transport Signalling Gateway (T-SGW)

This component in the Rel 4/5 network is PSTN/PLMN termination point for a defined network. It maps call related signalling from/to PSTN/PLMN on an IP bearer and sends it to/from the MGCF. Needs to provide PSTN/PLMN <-> IP transport level address mapping.

The further use of the elements supporting a VoIP session you will find Chapter 6.3 and 7.

4.3 Voice codecs

The effectiveness of voice codecs has vastly improved over the past few years. GSM networks now have 'enhanced fullrate' (better quality) and 'halfrate' (less bandwidth with almost the same quality) Codec types. The Third Generation Partnership Project has selected Adaptive Multi-Rate (AMR) as narrow band speech codec and the G.722.2 AMR-WB [14] as the wideband speech codec for UMTS Release 5 [15].

4.3.1 AMR and AMR-WB alternative codec schemes

AMR and AMR-WB were originally designed for circuit-switched mobile radio systems. Due to their flexibility and robustness, they are also suitable for other real-time speech communication services over packet-switched networks such as the Internet [15].

AMR

The AMR Codec is a multi-mode Codec that supports 8 narrow band speech encoding modes with bit rates between 4.75 and 12.2 kbps. The name narrowband codec come from the 8 KHz sampling frequency used in AMR. The speech encoding is performed on 20 ms speech frames. Therefore, each encoded AMR speech frame represents 160 samples of the original speech.

AMR-WB

Similar to AMR, the AMR-WB Codec is also a multi-mode speech Codec. AMR-WB supports 9 wide band speech coding modes with respective bit rates ranging from 6.6 to 23.85 kbps. The name wideband codec come from the 16 KHz sampling frequency used in AMR-WB. The speech processing is performed on 20 ms frames. This means that each AMR-WB encoded frame represents 320 speech samples.

Since this coding technique has compelling applicability to wired and wireless platforms, interoperation between 3G wireless infrastructure and fixed IP networks can be realized more easily.

Some common advantage

- Multi-rate encoding (i.e., multi-mode) capability of the codecs is designed for preserving high speech quality under a wide range of transmission conditions. The codecs can dynamically adjust the speech encoding rate during a session so as to continuously adapt to the varying transmission conditions. To perform adjustments, the receiver gives the sender a Code Mode Request (CMR).



- The codecs support Voice Activity Detection (VAD) and generation of Comfort Noise (CN) parameters during silence periods. Hence, the codecs have the option to reduce the number of transmitted bits and packets during silence periods to a minimum.

4.4 Protocols for real-time traffic in UMTS rel.5

To transport the voice packets in the UMTS network we need suitable protocols. Since the Release 5 of the UMTS is moving towards an “All-IP network”, the protocols will be much the same as for usual IP networks. Figure 4-4 presents the user plane protocol stack of a 3G PS conversational multimedia terminal explaining the transport of different media types.

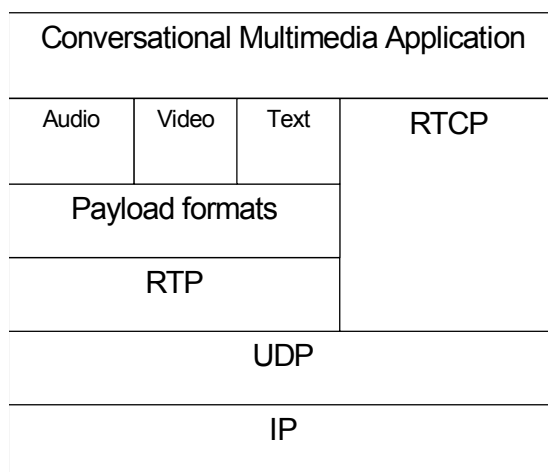


Figure 4-4: User plane protocol stack for 3G PS conversational multimedia terminal

[16]

4.4.1 Transport and control information protocols

The RTP protocol is used for media encapsulation for packet switched conversational multimedia applications within the 3GPP IM Subsystem. The RTP control protocol is based on the periodic transmission of control packets to all participants in the session, using the same distribution mechanism as the data packets. The underlying protocol must provide multiplexing of the data and control packets, by using separate port numbers with UDP. This result in packets encapsulated using RTP/UDP/IP in UMTS release 5, and will contribute 40 bytes of overhead to each packet. Thus, at a packetization interval 20 ms, headers alone would generate 16 kbps, too much for slow serial links. New compressing techniques have possibilities to cut the header down to 4-6 bytes.

As described in Chapter 2.6.1, RTP is not adequate protocol to support a real-time transfer. The data transport is augmented by a Real-Time Control Protocol (RTCP) to



allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality. RTCP packets convey end-to-end information about the quality of the session to each participant. Quantities like packet delay, jitter, and packets received and lost, are very valuable for the network to use to assess its own health in real time.

RTCP performs four functions [6]:

- The primary function is to provide feedback on the quality of the data distribution.
- RTCP carries a persistent transport-level identifier for an RTP source called the canonical name or CNAME. CNAME keep track of each participant and is also require by the receivers to associate multiple data streams from a given participant in a set of related RTP sessions, for example to synchronize audio and video.
- The first two functions require that all participants send RTCP packets, therefore the rate must be controlled in order for RTP to scale up to a large number of participants. By having each participant send its control packets to all the others, each can independently observe the number of participants. This number is used to calculate the rate at which the packets are sent.
- A fourth, optional function is to convey minimal session control information, for example participant identification to be displayed in the user interface. This is most likely to be useful in "loosely controlled" sessions where participants enter and leave without membership control or parameter negotiation. RTCP serves as a convenient channel to reach all the participants, but it is not necessarily expected to support all the control communication requirements of an application. A higher-level session control protocol may be needed.



4.4.2 Signalling protocol

3GPP has chosen SIP Session Initiation Protocol as the main signalling protocol for the UMTS Release 5 networks, to offer traditional telephony and enhanced multimedia services. SIP is used for signalling between the UE and the IMS as well as between the components within the IMS. It has been chosen because of its simplicity, extensibility and its wide availability.

UMTS SIP is interfaced with conventional call setup protocols used in mobile and landline telephone networks [17]. SIP supports:

- User location that is determination of the end system to be used in communication
- User capabilities, which means determination of the media parameters to be used in communication.
- User availability, the willingness of the called party to engage in communications
- Call setup, which means establishment of call parameters at both called and calling party.
- Call handling, which means transferring and terminating calls.

Somewhat simplified one can see SIP as a way to find the called party, agree on parameters and establish the call.

To resume this chapter, the implementing of VoIP in UMTS Rel 5 are done like this: The individual voice calls are independently encoded and packetized to appropriate separate Real Time Protocol (RTP) packets. These packets are then controlled by the RTCP and transported end-to-end inside UDP datagrams over real-time IP connections that have been negotiated and opened between the terminals during a SIP establishment. In Chapter 6 there are further explanations on SIP establishments and how to implement the QoS needed for reliable transfer.



5 An approach to IP QoS

In this report, we think of QoS as the quality VoIP experiences in an IP network. In VoIP environment it is important to minimize the latency experienced by the packet transporting digitized voice. To do so requires the use of a more modern and evolving standards that provide what is referred as to quality of service. In the simplest sense QoS means providing consistent, predictable data delivery service and satisfying customer application requirements.

5.1 QoS parameters and management

A number of factors are taken into account when specifying QoS in usual IP network. Here we also give a practical evaluation/accommodation on the factors to fulfil the voice requirements in a network.

- **Latency** – a synonym for *delay* is an expression of how much time it takes for a packet of data to get from one designated point to another. There are accommodated that the one way delay not becomes greater than 250 milliseconds, because then the talker overlap becomes a problem. When the round-trip delay becomes greater than 50 millisecond echo perceives as a problem [5].
- **Jitter** – is the deviation in or displacement of some aspect of the packets in a transmission. Timing variations is specially damaging for multimedia applications that rely on packets in a correct order at a given time. When packets are misplaced or late, the user can experience blurry images in real-time video applications.
- **Throughput** – the amount of data moved successfully from one place to another in a given time period. This reflects the bandwidth of the network and is a significant factor to QoS for e.g. VoIP. The requirements depend on the coding and compressing techniques.
- **Packet loss** – the percentage of packets lost in the transmission. Different applications will have different tolerance of packet loss. The loss does not severely affect voice fidelity if the lost packets are less than approximately 5 % of the total packets transmitted [18].

The above factors serve as reasons to introduce QoS in a network, especially when we are discussing QoS for real-time traffic. We have several mechanisms for dealing with these factors. QoS is split into two management architectures; IntServ and DiffServ.



5.2 Integrated Services

The Integrated Services (IntServ) [19] architecture provides a means for the delivery of end-to-end Quality of Service to applications over heterogeneous networks. To support this end-to-end model, the IntServ architecture must be supported over a wide variety of different types of network elements. The Internet Engineering Task Force has established one of two groups to develop Integrated Services QoS standard. The focus is on Integrated Services is long lived unicast and multicast flows, and the Resource Reservation Protocol (RSVP) [20] is used with this approach. Since RSVP and Integrated Services are strongly connected we will describe these together.

5.2.1 Integrated Services framework

The Integrated Services framework gives applications opportunity to select from multiple, controlled levels of QoS for their packets. To assure End-to-End QoS, all network nodes have to support mechanisms to control the desired QoS. There must also be a way to communicate the application's requirements to the routers.

QoS in the Integrated Services framework is provided by two Network Element Services; Controlled-Load and Guaranteed-Service. These services need to be set up by either an application or the network itself. There are multiple possibilities for communication between applications and routers, but is in most cases implemented by the resource reservation setup protocol named RSVP. Figure 5-1 illustrates the framework of IntServ and RSVP.

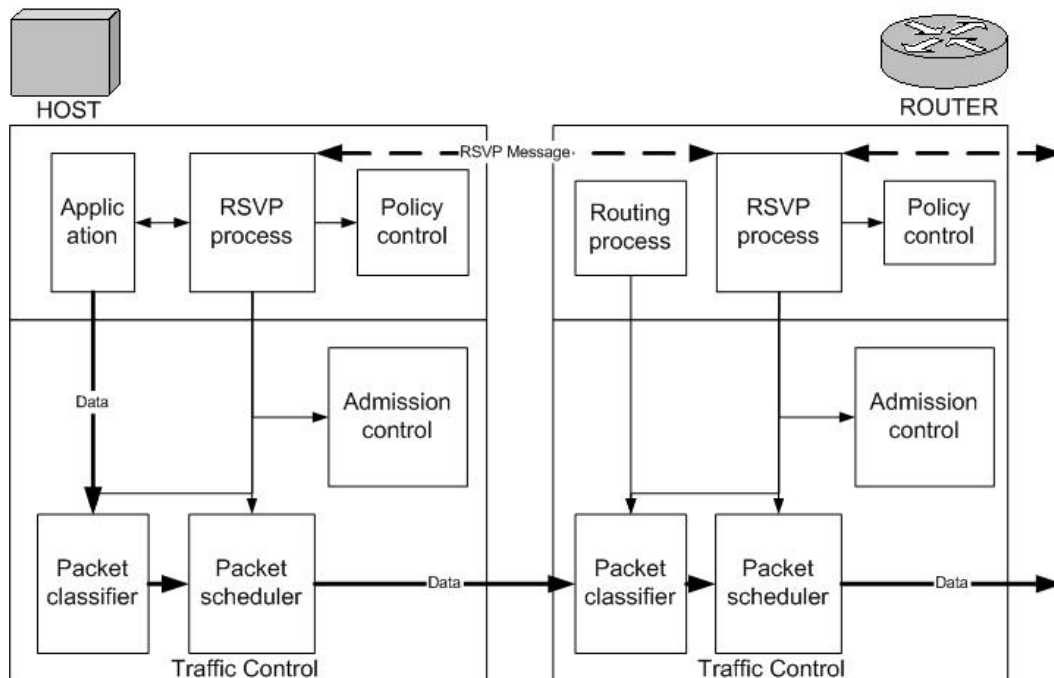


Figure 5-1: IntServ framework



Quality of service is implemented for a particular data flow by mechanisms collectively called "traffic control". All incoming packets are fed to the packet classifier to determine the QoS class and the route of the packets. The packet scheduler queues the packets on behalf of the determined parameters set, and forwards the packets according to the promised QoS.

In all IntServ QoS-capable nodes, the admission control and policy control modules control resource reservation requests. The admission control module makes sure that there are sufficient resources available for a required request, and the policy control module determines if the sender is allowed to make a reservation on the network node.

5.2.2 RSVP

Source applications supporting IntServ uses the RSVP to tell the other party about its upcoming flow. The receiver makes the actual reservation after the sender has inquired a session. There are other reservation protocols available for IntServ, but RSVP is the one mostly used. RSVP should not detract from the robustness that we find in today's connectionless networks. Because connectionless networks rely on little or no state being stored in the network itself, it is possible for routers to crash and reboot and for links to be active or inactive while end-to-end connectivity is still maintained. In this matter, we find that RSVP solves this by using the soft state in the routers. Soft state- in contrast to hard state found in connection-oriented networks does not need to be explicitly deleted when it is no longer needed, but is removed when it is not periodically refreshed.

An application wishing to make a reservation will initiate a QoS session by sending a PATH message to the other party. This message stores "path state" in the routers on the way and contains at least the unicast IP address of the previous hop node to preserve the reverse route from the receiver, back to the sender.

A PATH message is required to carry a Sender Template, Sender Tspec and an Adspec. Sender Template describes the format of the packets the sender will originate. Tspec defines the traffic characteristics of the data flow that the sender will generate. Adspec carries advertisement information about the senders QoS capabilities.

Each router looks at the PATH message as it goes past, and it figures out the reverse path that will be used to send reservations from the receiver back to the sender in an effort to get the reservation to each router on the path. Having received a PATH message, the belonging application will calculate the needed reservations on behalf of the contents of the message. A RESV message will be generated and sent back up towards the sender.

This message contains the requirements of the receiver. Each router on the path looks at the reservation request and tries to allocate the necessary resources to satisfy it. If the reservation cannot be made, an error message is returned to the receiver; else the RESV message will be passed on to the next router. All routers will do this check, and if all goes well, the correct reservation is installed on each one of them, ending up with the sender host, making the reservation complete. The host then sets up the appropriate traffic



control parameters. It is important to notice that the demand specified in the PATH message not necessarily equals the demand received in the RESV message. The receiver decides whether or not it can use the preferred bandwidth requested from the sender.

5.2.3 Service classes

- **Controlled Load**

Controlled Load [21] offers a service quality that approximates the behaviour visible for applications receiving best effort service under unloaded conditions from network elements. In a correctly working network, these applications may assume that a high percentage of the transmitted packages will be successfully delivered at the network end nodes, at the same time as the delay does not greatly exceed the minimum transit delay.

To ensure a controlled load service, the clients requesting the service will have to provide the intermediate network elements with an estimation of the traffic generated (TSpec). Should the client's traffic generating properties fall outside the region described by TSpec parameters, the client will suffer from delays and packet loss. It is important to notice that Controlled Load gives an assurance on behalf of the assumptions taken by the applications, but no guarantees.

- **Guaranteed Service**

Guaranteed Service [22] differs from Controlled Load by guaranteeing a maximum transmission delay on behalf of the requesting limit. This service is intended for applications vulnerable for late arriving packets, such as real time voice and video applications.

The guaranteed service is implemented by two sets of parameters; traffic parameters (TSpec) and service-level parameters (RSpec). RSpec defines the quality requested and consist of data rate and slack term. In addition, two error terms describe the accuracy of the reservation compared to a perfect one. Taking advantage of combining the parameters from the various service elements in a path gives us the opportunity to obtain the maximum delay in the transmission. This method is complicated and will not be discussed further at this time.

Applications using Guaranteed Service in a QoS control achieves considerable control over their delay. Delay is split in two parts; fixed delay due to the given property of the path, plus delay experienced from queuing in the routers. Guaranteed Service can control the queuing and implies in that applications can accurately estimate, in advance, what queuing delay Guaranteed Service will promise. If the delay assumption differs greatly from experienced delay during transmission, the token bucket and data rate can be changed and optimised to achieve a lower delay.



5.3 Differentiated Services

DiffServ was designed to provide a simpler, coarser approach to establishing differentiated classes of service for Internet data. In the IntServ model, resources are allocated to individual flows, which can lead to scalability limitations. In DiffServ, traffic is divided into a small number of forwarding classes, and resources are allocated on a per-class basis. The desired performance levels are achieved through the proper mix of provisioning, prioritisation, and admission control. This is in contrast to techniques such as end-to-end resource reservation, which is the foundation of RSVP. Although others are possible, there are currently two standardized per hop behaviours (PHBs) defined that effectively represent two service levels (traffic classes); Expedited Forwarding and Assured Forwarding.

5.3.1 The Differentiated Services Framework

DiffServ compliant nodes are put into an architecture that makes the core network simple with few complex functions. Complex functions are put into the end nodes. This is what makes the DiffServ scalable.

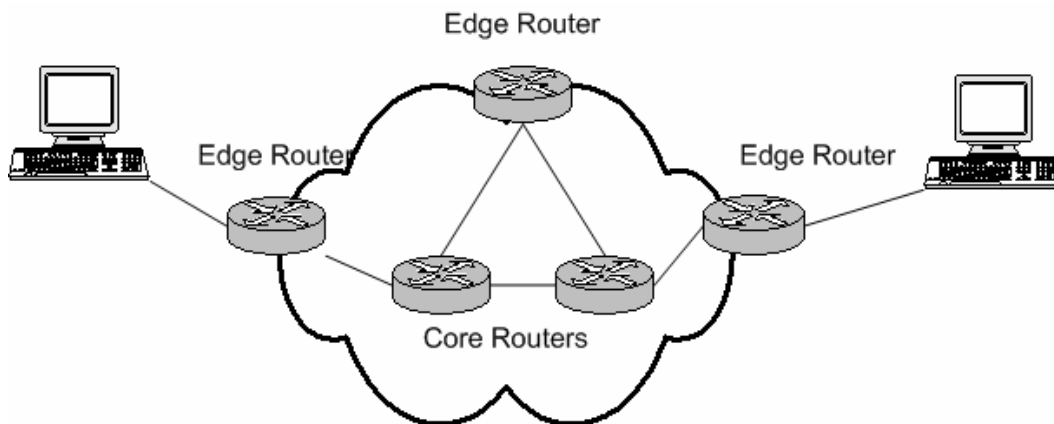


Figure 5-2: Differentiated Services architecture

Figure 5-2 shows the architecture in few details, it is adequate to explain the basic functions for a DiffServ network. The architecture is based on a simple model where traffic entering a network is classified and possibly conditioned at the boundaries of the network, and assigned to different behaviour aggregates (BA). The Traffic conditioner marks each packet with a Differentiated Service Code Point (DSCP) given by an admission system. No signalling is necessary for differentiated services throughout in the network since traffic conditioning is configured to satisfy the user contracts.

Core routers are located inside the network and provide differentiated services by different scheduling classes called Per-Hop-Behaviours (PHBs). The DSCP value set by the edge router is used to determine the suitable PHB.



5.3.2 DiffServ Domain

A DiffServ Domain (DS Domain) is a contiguous set of DiffServ compliant nodes with a common service provisioning policy and set of PHB groups implemented on each node. A DS domain consist of one or several networks under the same administration responsible for ensuring that adequate resources are available to support services offered by the domain.

A DS domain has a well-defined boundary with boundary nodes which deals with classification and conditioning of ingress traffic to ensure that transit traffic is marked appropriately to select a supported PHB within the domain. Edge routers surrounding a DS Domain are known as Boundary nodes where traffic entering a domain at DS Ingress node and exits at DS Egress nodes. Inside the network, traffic is forwarded through Interior Nodes. Interior nodes inspect only one field in the IP header-the DiffServ field-to determine where to send the packet next. In the scenarios where the end-to-end paths expands over several DS domains, called DS region, the boundary egress node will remark and shape the packets to suit the next DS domain properly. If the user does not have agreements with other than the first DS domain, the packets could end up given a poorer PHB.

5.3.3 Differentiated Services Field

In a DiffServ network, we want the boundary routers to mark incoming packets so that an aggregate can expect a certain forwarding treatment through a DS domain. Each packet entering a DS domain will be assigned a value called a Differentiated Service Code (DSCP) [23] in the IP header. Figure 5-3 shows the IPv4 TOS (Type Of Service) header and how the Differentiated Service Field [23] is meant to work.

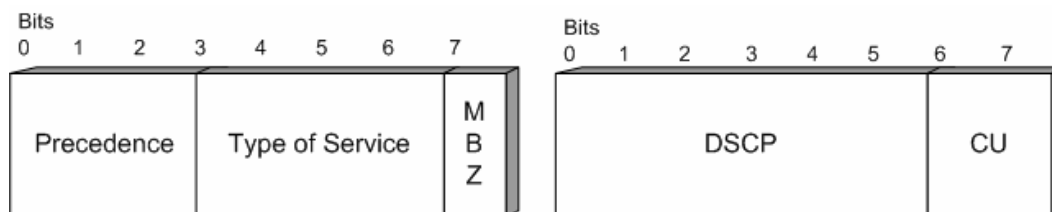


Figure 5-3: Differentiated Services Code Points redefine IP precedence IPv4 TOS byte

A DSCP value is a reference to which forwarding treatment the aggregate would prefer through a DS domain. Within a DS domain can many individual application-to-application flows share the same DSCP value if the flows is meant to be treated the same. The collection of packets sharing the same DSCP value is referred to as a Behaviour Aggregate (BA) and the core routers will only need to take BAs in concern rather than particular flows. A DS domain however is free to use non-standard DSCPs within the domain as long as packets are remarked when leaving the DS domain.

The DSCP field is set to 6 bit, making it possible to separate 64 different classes of forwarding treatment, Per Hop Behaviour (PHB). In IPv4 TOS field, there is already a



function for priority traffic implemented, which is called IP Precedence. Since there may be users using IP Precedence field (bits 0-2), DiffServ should be implemented backward compliant to prevent failures in an early process. A Class Selector is implemented to remark the IP Precedence packets to an equivalent PHB.

5.3.4 Per Hop Behaviours (PHBs)

A Per-Hop Behaviour gives an aggregate a behaviour characteristic. The packets passing through a DS node will carry a decided DSCP value and indicates which treatment the DS nodes will give them on a per-hop basis. The treatment is realized by a number of queuing and scheduling algorithms which all have different characteristics whether we focus on jitter, delay or bandwidth. We call a set of nodes with closely related behaviours for a PHB group. A DS node can maximum handle 64 different PHBs, the same amount of DSCP combinations available.

- **Expedited Forwarding (EF)**

Expedited Forwarding [24] is the easiest PHB to explain. Packets marked for EF treatment should be forwarded with minimal loss, latency and jitter, the three factors which we experience due to queuing transiting the network. To guarantee little or no queuing, an aggregates maximum arrival rate has to be less than the same aggregates minimum departure rate.

The rate limiting of EF packets can be achieved by configuring the ingress node of a DS domain to allow a certain maximum rate for incoming EF packets. A simple but conservative approach would be to ensure that the sum of the rates of all EF packets entering the domain is less than the bandwidth in the slowest link in the domain. Even in worst case, when all EF packets converge on the slowest link, the network will avoid being congested and can provide the desired behaviour.

Several types of queue scheduling mechanisms may be employed to deliver the forwarding behaviour EF. The RFC2598 [24] does not specify a particular approach, but points out the fact that the mechanisms suitable for EF have different jitter and delay characteristics. When implementing a DiffServ EF, the queue scheduling shall make sure that there is little or no queue visible for EF traffic, resulting in a low loss, low latency and low jitter service, which are suitable for real time human communication such as packet telephony.

- **Assured Forwarding (AF)**

Besides the EF PHB, which supports services with strict demands to bandwidth and jitter, we have another called Assured Forwarding (AF) [25]. AF support four levels of PHB, where each of them is determined to assure a flexible and dynamic sharing of the network with their own values of buffer space and bandwidth. The focus is set to the soft bandwidth and loss guarantee suitable for bursty traffic.



Within each AF class IP packets are marked with one of three drop precedence values. The drop precedence informs the router about the importance of a packet being delivered in case of congestion. The level of forwarding assurance of an IP packet thus depends on forwarding resources allocated to the AF class, the current load of the class, and in case of congestion, what the drop precedence is set to.

All the complex control functions in an AF PHB group is set and controlled by the edge routers. The core routers make sure that the requirements set by the edge routers are followed.

The management and mechanisms for supporting QoS in UMTS network are described in Chapter 6.3. In Chapter 7 we will describe how fixed IP network can cooperate with UMTS network to give reliable VoIP sessions.



6 Quality of Service in UMTS Release 5

The implementation of VoIP sessions in UMTS Release 5 demands a reliable QoS. We will present the four QoS UMTS classes, and the important parameters for a reliable voice call. Further on, in Chapter 6.3, we will focus on the QoS management in the UMTS network.

6.1 QoS classes in UMTS

When defining the UMTS QoS Classes, also referred to as traffic classes, the restrictions and limitations of the air interface has to be taken into account. It is not reasonable to define complex mechanisms as have been in fixed networks due to the different error characteristics of the air interface.

There are four different QoS classes standardized by 3GPP [26]:

- Conversational class
- Streaming class
- Interactive class
- Background class

The main distinguishing factor between these QoS classes is how delay sensitive the traffic is. Conversational class is meant for traffic which is very delay sensitive, and background class is the least delay sensitive class.

Conversational class

The most well-known use of this class is telephony speech like GSM. Internet and multimedia applications will require this scheme, such as voice over IP and video conferencing tools. Real time conversation as always performed between human end-users, and makes this scheme the only one where the required characteristics are strictly given by human perception.

Conversational class – fundamental characteristics for QoS:

- The time relation (variation) between information entities of the stream has to be preserved.
- Conversational pattern have to be stringent with low delay.

Streaming class

When a user is watching real time video, the streaming class is used. The data flow is always a one-way transport, aiming at a live human destination.

Streaming traffic – fundamental characteristics for QoS:

- Preserve the time relations (variation) between information entities within a flow, but with tolerance of delays.



Interactive class

The interactive scheme applies when end users, either a machine or a human, requests data from remote equipment such as servers. Examples of human interaction with the remote equipment are: web browsing, data base retrieval, server access. Examples of machines interaction with remote equipment are: polling for measurement records and automatic data base enquiries.

This type of traffic follows a pattern of request-response. The end user sends a request to the remote equipment, and is dependent on a response within a certain time. Round trip time is therefore one of the key parameter in addition to expected transparently transferred packets (with low bit error rate).

Interactive traffic – fundamental characteristics for QoS:

- Request response pattern
- Preserve payload content

Background class

The background scheme applies when end users, typically a computer, sends and receives data-files in the background. Examples of background traffic are E-mail delivery, SMS, download of databases and reception of measurement records.

The destination is not expecting the data within a certain time, thus more or less delivery time insensitive, but the background traffic shall be transparently transferred with low error rate.

Background traffic – fundamental characteristics for QoS:

- The destination is not expecting the data within a given time
- Preserve payload content



6.2 QoS parameters and requirements

The following table shows the most important QoS parameters for a voice call. As we can see the parameters do not vary much from the generally IP networks.

Table 6-1: Parameters needed for an end-to-end voice transfers in UMTS [11].

Characteristic	Generally	With UMTS
Data rate	4 kbps - 25 kbps	4,75 kbps - 12,2 kbps
One way delay	< 150 ms (max. limits = 400 ms)	
Variation of the delay (Jitter)	< 1 ms	
Information loss	< 3 % framework loss rate	SIR ~ 5 dB

It is important to understand that the end-users only care about issues that are visible to them. This involves some requirements/statements:

- Only the QoS perceived by end-user matter;
- The number of user defined/controlled attributes has to be as small as possible;
- Derivation/definition of QoS attributes from the application requirements has to be simple;
- QoS attributes shall be able to support all applications that are used, a certain number of applications have the characteristic of asymmetric nature between two directions, uplink/downlink;
- QoS definitions have to be future proof;
- QoS has to be provided end-to-end.

In Chapter 8, we have done some practical testing of the UMTS network with the concern on the factors in Table 6-1.



6.3 Management of QoS in the UMTS network

In regards to IP QoS we have discussed how the protocols work and how QoS is obtained, but the UMTS network has a unique way of negotiating QoS. Simplified, UE use SIP signalling to send QoS parameters required and the IMS will, on behalf of the required QoS, make a decision whether or not the resources are available. The QoS negotiating is split in two; Authorization for resources and reservation of the authorized parameters. These procedures handle the QoS negotiation in scenario 2, described in Chapter 2.2 VoIP scenarios.

6.3.1 Authorization of QoS resources

When a subscriber wants to initiate a session with QoS, the first step will be to inquire an authorization for QoS. The P-CSCF (PDF) use the SDP (Session Description Protocol) contained in the SIP signalling to calculate the proper authorization, see Figure 6-1 [27]. SDP is purely a format intended for describing multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation.

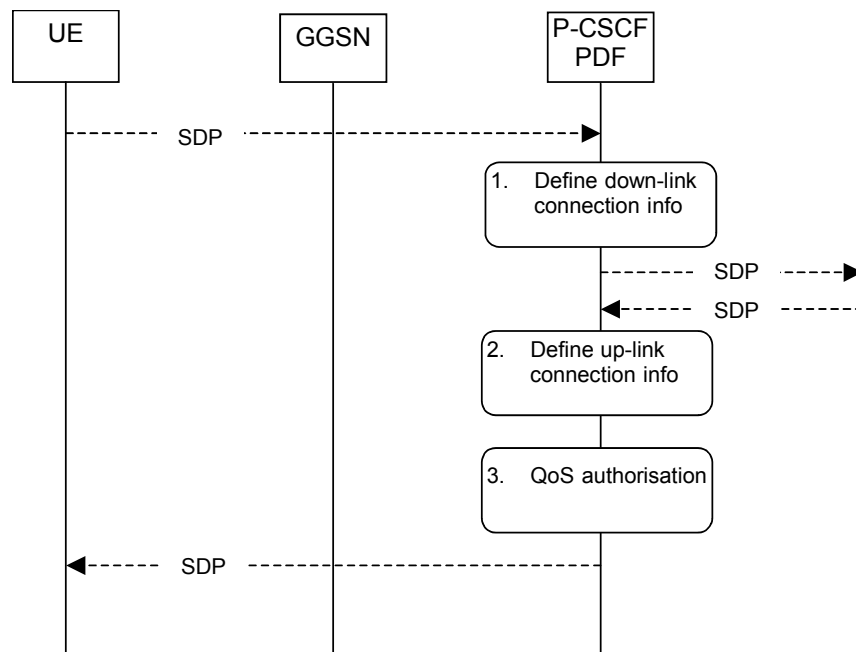


Figure 6-1: Authorize QoS resources at originating PDF

The P-CSCF(PDF) gets the SDP parameters defined by the originator and identifies the connection information needed, such as IP address of the down link IP flow(s), port numbers to be used etc. The P-CSCF (PDF) gets the negotiated SDP parameters from the terminating side through SIP signalling interaction. The P-CSCF (PDF) identifies the connection information needed for the downlink. The P-CSCF (PDF) uses the SDP



parameters in order to define the QoS resource authorization. The PDF authorizes every component negotiated for the session, and shall be expressed in terms of IP QoS parameters. An authorization token is generated by the PDF and sent to the UE.

When UE has received a token proving the right to obtain QoS, a reservation of the negotiated QoS falls into place.

6.3.2 Resource reservation

The GGSN serves as the Policy Enforcement Point (PEP) that implements the policy decisions for performing admission control and authorising the GPRS and IP BS (Bearer Service) QoS Resource request, and policing IP flows entering the external IP network.

Before an MS can access a service, the *Packet Data Protocol (PDP) context* for the service must be activated, which specifies the packet data protocol and the routing method used for the communication session. The PDP context is maintained in the UE, the SGSN, and the GGSN. After successful QoS authorisation, the UE will perform a resource reservation for a flow with a PDP context request, see Figure 6-2. Whenever an IP address is needed, a Packet Data Protocol (PDP) context needs to be established. A PDP context contains PDP type (IPv4 or IPv6), PDP address assigned to the terminal, the requested QoS parameters and the GGSN address.

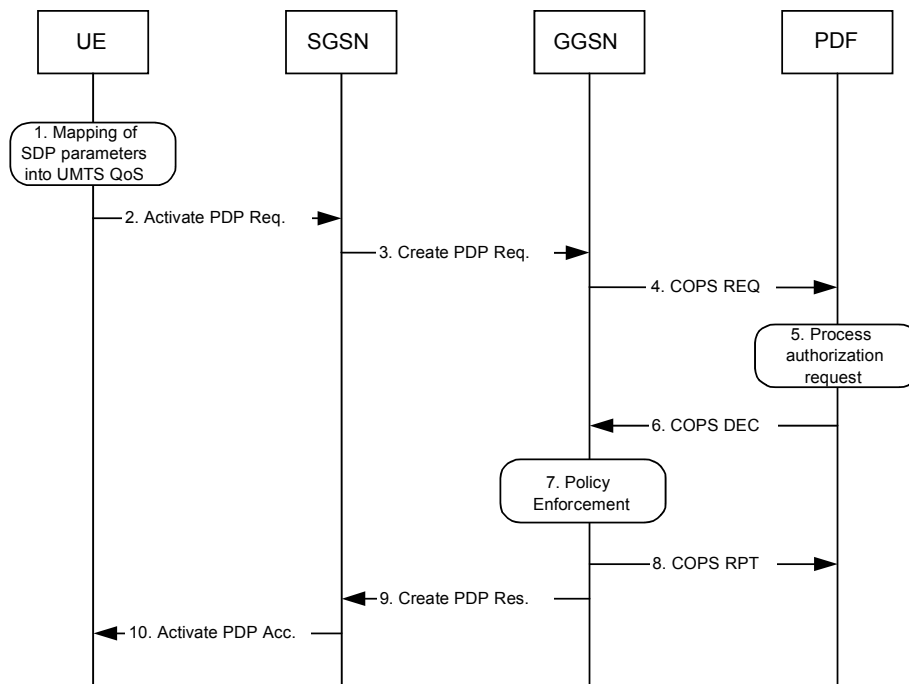


Figure 6-2: Resource reservation flow with service based local policy



1. The UE uses the SDP parameters to define the UMTS QoS parameter needed to request a PDP context. A mapping from SDP parameters to UMTS QoS parameters is needed to send a PDP request to SGSN. This mapping mechanism will not be discussed further due to complexity and is insignificant for this thesis.
2. The UE sends an “Activate PDP request” containing binding information; authorization token generated by P-CSCF (PDF) and mapped QoS parameters.
3. The SGSN will forward this message with a “Create PDP Request” to the GGSN. The correct PDF is found by analysing the authorization token.
4. GGSN request an integrity check from the PDF so that the IP QoS parameters match the authorized QoS.
5. The PDF identifies the multimedia session by using the binding information and performs an authorization decision.
6. The result from the match is send with a “COPS DEC” message back to the GGSN.
7. If the QoS parameters are approved, the GGSN will map the parameters to PDP QoS parameters and accept the PDP activation. If the requested QoS parameters are not within the authorized QoS, the GGSN downgrades the requested UMTS QoS parameters.
8. GGSN sends COPS RPT message back to the PDF and reports its success or failure in carrying out the PDF decision.
9. GGSN accepts the PDP context request based on the results of the authorization policy decision enforcement. If the requested QoS parameters are not within the authorized QoS, the GGSN downgrades the requested UMTS QoS parameters.
10. The SGSN sends an Activate PDP Context Accept message to the UE indicating that the PDP context is activated and that the QoS requirements have been authorized for both up- and downlink.

When the PDP context is registered, the UE has initiated a session for transport with the authorized QoS parameters. Each flow is put into a traffic class with belonging characteristics. If the application running requires a real time flow, an adequate traffic class is selected. Treatment of packets in GGSN is further described in Chapter 7.2. A notice to this initiating process is that it determines the QoS inside the UMTS network, not in fixed public networks.



7 UMTS interworking with fixed networks

The “all IP” architecture is required to provide interworking solutions between UMTS and 2nd Generation mobile networks, the legacy PSTN networks and external IP-based SIP/ H.323-enabled networks.

7.1 Overview

Figure 7-1 shows the paths UMTS to PSTN and UMTS IP to IP. Note that we use the same set of scenarios as in Chapter 2.2 VoIP scenarios.

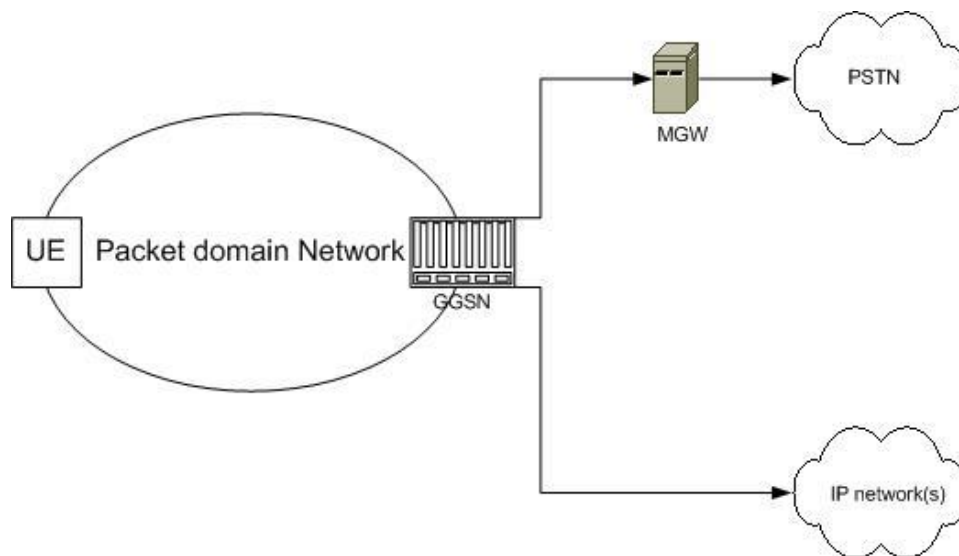


Figure 7-1: External networks interworking

Scenario 3: IP to PSTN: A UMTS terminal makes a call to a user located in the PSTN circuit switched network. The UMTS terminal sends VoIP packets to the IMS system where they are translated into traditionally TDM frames. The logic node in the IMS responsible of this function is called a Media Gate Way (MGW), see Figure 7-1.

Scenario 1: UMTS IP to IP: When interworking with the IP networks, the Packet Domain can operate IPv4 or Ipv6. The interworking point with IP networks is at the Gi reference point as shown in Figure 7-1. The Gi reference point is located between the GGSN and MGW and between GGSN and multimedia IP networks. The IMS plays an important role concerning IP to IP as well. There is no support for call control and setup signalling in early stages of UMTS but will be provided by UMTS release 5.



Typically in the IP networks, the interworking with sub networks is done via IP routers. From the external IP network's point of view, the GGSN is seen as a normal IP router. In relation to QoS and IP network interworking, we will discuss the GGSN further.

7.2 IP network interworking QoS management

The GGSN is located in the core network (switching network) and act as a gateway in GPRS and UMTS networks. The GGSN coordinates with the SGSN to exercise the QoS management function. The GPRS Tunneling Protocol (GTP) between the GGSN and the SGSN does not provide the QoS function. Instead, the QoS of user data packets is supported by lower layer. Figure 7-2 shows the QoS architecture of the GGSN [28].

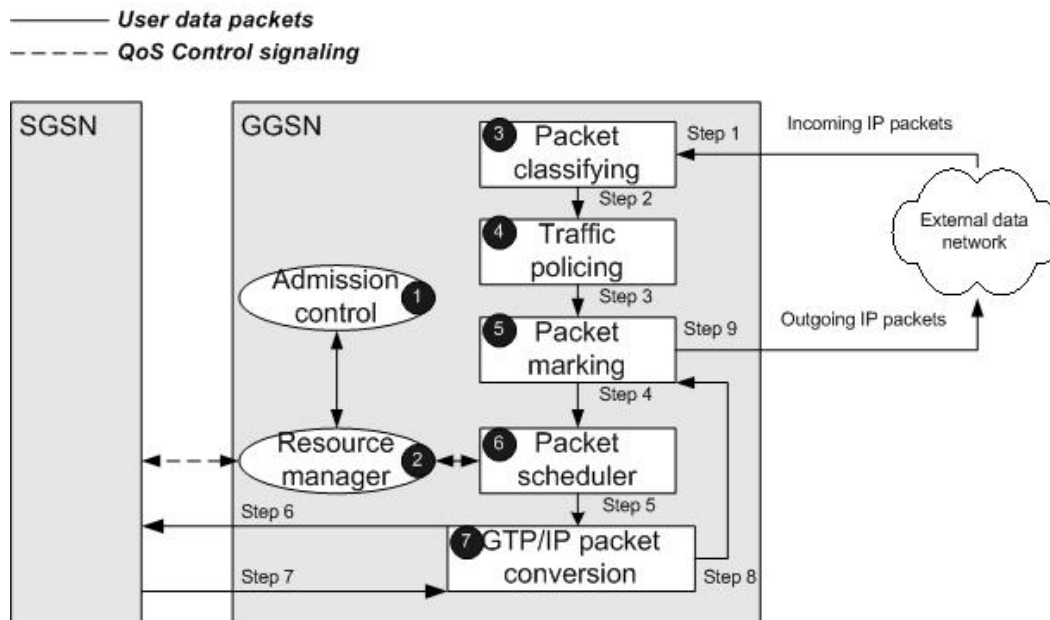


Figure 7-2: GGSN QoS architecture

The QoS management in the GGSN includes the following functions (numbered with black circles in Figure 7-2).

1. **Admission Control** determines if a new or modified PDP context request can be accepted based on the available resource.
2. **Resource Manager** provides the QoS parameters required for each individual bearer service, which is responsible for *resource reservation* and *resource monitoring*. Resource reservation reserves negotiated resources for the PDP context of a real-time application. Resource monitoring is responsible for tracking the available system resources.



3. **Packet Classifying** maps each downlink user packet to the corresponding PDP context using the *Traffic Flow Template* (TFT). Packet classifying enables multiple PDP contexts to share one IP address. Specifically, TFT employs IP header fields and higher level headers (UDP/TCP) to differentiate each PDP context.
4. **Traffic Policing** forces a user's traffic to conform to the reserved resources. If the incoming user data traffic exceeds the maximum bit rate specified in the corresponding PDP context, the user data packets are dropped to restrict the user traffic within the allowed limit.
5. **Packet Marking** marks each data packet with specific QoS indication related to the bearer service for data packet transfer.
6. **Packet Scheduler** determines the transfer priority of incoming data packets based on their QoS classes. For example, real-time traffic (conversational and streaming classes) has higher priority over that of non-real time traffic (interactive and background classes).

We will explain how a packet is treated by the GGSN in a QoS session. We will not explain signalling in this context. We expect that the desired QoS is already negotiated and ready for use. Note that the Admission Control is involved in PDP context activation explained in Chapter 6.3.2. The Packet Classifying, Traffic Policing, Packet Marking and Packet Scheduler are involved in packet delivery. The Resource Manager is involved in both PDP context activation and packet delivery. The QoS processing for an incoming IP packet from the external data network to the SGSN is illustrated in Figure 7-2, and is described as follows.

- Step 1:** The incoming IP packet is categorized by the Packet Classifying based on the corresponding PDP context.
- Step 2:** The Traffic Policing examines if the incoming IP packet traffic exceeds the allocated bandwidth. If so, the IP packet is dropped.
- Step 3:** Based on the PDP context, the Packet Marking examines and marks the IP packet with corresponding QoS parameters.
- Step 4:** Based on the priority class the, Packet Scheduler checks available resources with the Resource Manager to determine the delivery schedule for the IP packets. For example, the delivery priority of the streaming class traffic is higher than that of the background class traffic. If the current available resource cannot support delivery of all packets, the Packet Scheduler may queue the packets of low priority traffic (such as interactive and background class traffic) and wait for the resource.
- Step 5:** The GTP/IP Packet Conversion encapsulates the IP packet into a GTP packet.



Step 6: Finally, the GTP packet is transferred to the SGSN.

The QoS processing for an outgoing GTP packet from the SGSN to the external PDN is described as follows:

Step 7: The GTP packet is decapsulated to an IP packet by the GTP/IP Packet Conversion.

Step 8: The Packet Marking marks the IP packet with the QoS parameters of the external PDN by mapping the QoS parameters of UMTS into those of the external PDN. For example, if the external PDN supports the DiffServ QoS mechanism, the conversational class packets are marked with DiffServ *Expedited Forwarding* code point. The streaming and interactive class packets are marked with *Assured Forwarding class 1* and *class 2* code points, and the background class packets are marked with *Best Effort* code point.

Step 9: The IP packet is transferred to the external PDN.

This process highlights the treatment given to a packet belonging to a specific QoS traffic class. As mentioned in step 8, the GGSN have to provide QoS for the external network by mapping the UMTS traffic class preferences into IP QoS. Outside the UMTS network, the operators can make contracts with Internet Service Providers (ISPs) to be able to use a certain QoS procedure.

In Chapter 5 we explained two standardized QoS procedures; IntServ and DiffServ which is specified for use in IP networks. A combination of IntServ and DiffServ arise opportunities for using the benefits from both [29].



8 Testing the UMTS network

Testing is an important task in our thesis. We can then get values to compare with the theoretical values needed for accomplish a VoIP session.

8.1 Overview

The main focus was to test the performance of VoIP, especially between a node on UMTS network and a node on the Public Internet. We could then get indication on how the UMTS network worked together with a usual IP network. At Økern in Oslo there is a UMTS network that belongs to NetCom. In cooperation with Siemens we were able to do some tests on the network. The network is a trail network where IMS is not present.

8.2 Method

We used the program Chariot v4.3 to test different transfer sessions. We decided to use Chariot to do tests with the purpose of the modules available. Chariot has its own module for VoIP sessions, which became useful. Variables important for VoIP sessions are delay, jitter, throughput and Mean Opinion Score (MOS). The theoretical values for delay, jitter, and throughput needed for a transfer in UMTS, you will find in Table 6-1. MOS is directed directly to VoIP and is a rating system devised by the telephony industry to assess the customer's satisfaction with a service. With these variables we can evaluate the connection established in the desirable scenarios.

Table 8-1: Mean Opinion Score estimates

Mean Opinion Score (lower limit)	User Satisfaction
4.34	Very satisfied
4.03	Satisfied
3.60	Some users dissatisfied
3.10	Many users dissatisfied
2.58	Nearly all users dissatisfied

One PC maintained the administrator functions and with this we could set up the desired sessions with all the functions available in the Chariot Console Software. The Stationary workstation, Laptop A and Laptop B worked as endpoints. The configuration of the tests you will find in Figure 8-1. At the endpoints we installed the Chariot performance endpoint program. Performance Endpoints are lightweight software agents used by Chariot to collect information about network transactions and send it back for analysis and reporting. For more information, see the Appendix B "Chariot User Guide". The scenarios we determined useful are further described in Chapter 8.5 and 8.6. Many possibilities are made available working with VoIP in Chariot. We can select different parameters in a test like coding techniques, quality parameters and voice activity (VA). The coding techniques are represented as codecs, which will be discussed later. The important thing to note is that the one we choose to use, need the whole 64 kbps uplink bandwidth in the UMTS connection.



8.3 VoIP Codecs

Chariot 4.3 gives opportunity to test different kinds of VoIP codecs. Chariot supports a variety of codecs used in VoIP systems today, but have not yet thought about tests with mobile systems such as UMTS. Therefore, the codec G.722.2 suggested for wide band speech codec use in mobile networks is not available.

Table 8-2: Codecs supported by Chariot

RTP Payload Type (codec)	Coding Type	Data Rate in kbps	Data Bytes per 30ms packet	Total Data-gram Size (bytes)	Combined bandwidth for 2 flows (kbps)
G.711	PCM	64.0	240	298	158.93
G.726-32	PCM	32.0	120	178	94.93
G.729	CS-ACELP	8.0	30	88	46.93
G.723,1m	MP-MLQ	6.3	24	82	43.73
G.723,1a	ACELP	5.3	20	78	41.60

Table 8-2 shows the most common codecs used in VoIP networks today. Siemens envisions to optionally supporting AMR-WB, G.711, A-law, u-law and GSM 6.10. This information given by Siemens made us concentrate on the European G.711, meaning G.711 A-law.



8.4 Equipment architecture

There are several ways to test sessions with VoIP in a complex network such as UMTS, but not many paths open for free use. Restrictions made us exclude desirable scenarios because of security in the network. Figure 8-1 shows paths desirable for testing.

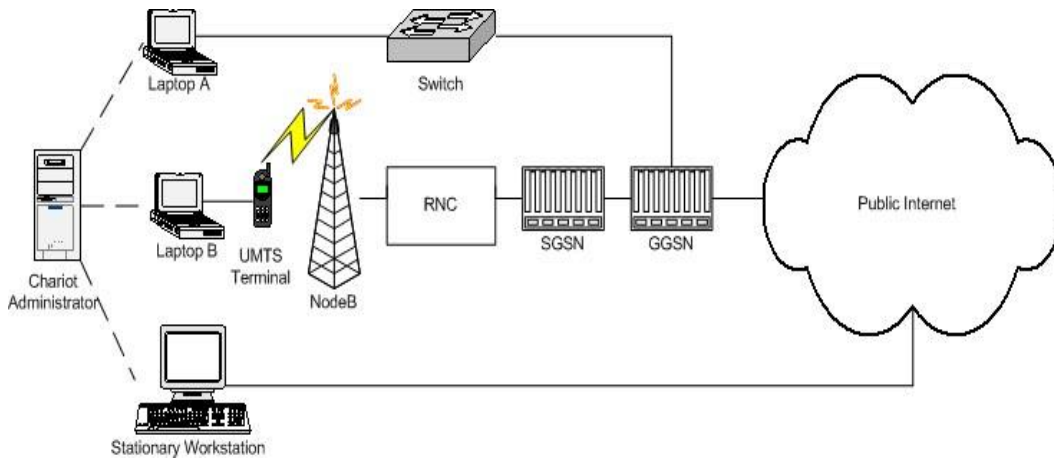


Figure 8-1: UMTS testing configuration

The scenarios we considered testing will now follow.

8.5 Scenario 1: Session between Laptop B and Stationary workstation.

Measuring between the UMTS 64/384 kbps to a stationary pc connected to a broadband channel gives us the overall delay when communicating in private IP networks. Further, we get a more realistic distance of the transfer. The transfer between a UMTS terminal and a PC on a LAN network connected to the public Internet or in private IP networks will be a popular sessions, especially for those eager to use a laptop PC for all kinds of communications.

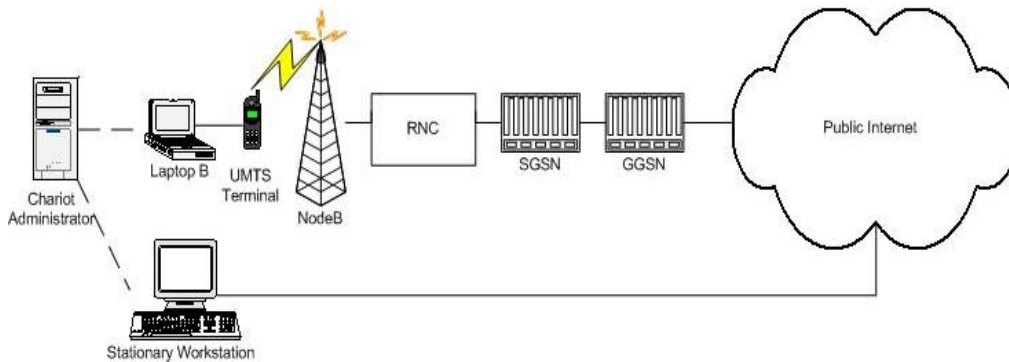


Figure 8-2: Scenario 1 model



8.5.1 Equipment involved in the test

- Laptop B (Pentium 2, 266 MHz, 128 MB RAM, Windows 2000)
- UMTS network by Siemens / NetCom
- Public Internet connected through NetCom ISP
- Workstation (Pentium 2, 266 MHz, 512 MB RAM, Windows 2000)
- Chariot administrator, another laptop (Pentium 3, 850 MHz, 256 MB RAM, Windows XP)

8.5.2 Geographical:

Laptop B was placed at Ulven, NetCom. The stationary workstation was placed at Agder University College in Grimstad.

8.5.3 Setup

In the Chariot Console program we set up a number of connections simulating calls made in the network. We tried to “fill the pipe” in a way where each UMTS customer will be provided with a 64 kbps channel. This is reasonable since a UMTS terminal dedicates a real number of 64 kbps channels. The test was divided in two parts; with and without QoS.



8.6 Scenario 2: Session between Laptop A and Laptop B

Measuring between the UMTS 64/384 kbps to a laptop pc connected to the same subnet as the UMTS terminal results in internal delays instead of public Internet delays. Both the Laptop A and B was situated at Ulven. The main difference was that the laptop B communicated via a UMTS terminal. This was a session on NetCom's internal network. It was important to get test results that we could use for further study. The comparison between data from scenario 1 and scenario 2 would be of great interests.

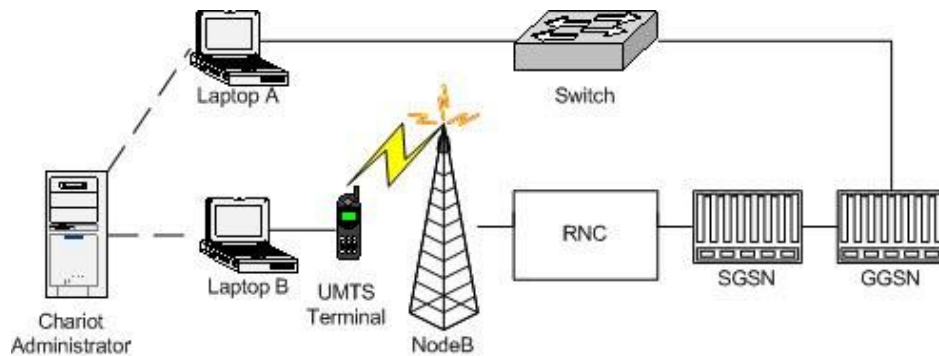


Figure 8-3: Scenario 2 model

8.6.1 Equipment involved in the test

- UMTS network by Siemens / NetCom
- Laptop A (Pentium 4, 1,7GHz, 256 MB RAM, Windows XP)
- Laptop B (Pentium 2, 266 MHz, 128 MB RAM, Windows 2000)
- Chariot administrator, another laptop(Pentium 3, 850 MHz, 256 MB RAM, Windows XP)

8.6.2 Geographical

Laptop A and Laptop B was placed at Ulven, NetCom.

8.6.3 Setup

In the Chariot Console program we sat up a number of connections simulating calls made in the network. We would try to “fill the pipe” in a way where each UMTS customer will be provided with a 64 kbps channel. This is reasonable since a UMTS terminal dedicates a real number of 64 kbps channels. The test would include voice activities from 50 to 100 %. The test was divided in two parts; with and without QoS.



9 Test Results

Problems occurred with scenario 1. In the GGSN there are several routers, and each of them can be set up in a different way. Due to security reasons, the TCP protocol was restricted for use outside the core network. Since the Chariot console program initialised the two devices used for measuring through TCP, we were unable to test scenario 1 by using the Chariot. This resulted in an impulsive and alternative test. We used the Tracer Route program located in Windows 2000 to get some results on the latency in a long distance conversation. Even though we came up with an alternative way of testing scenario 1, the Chariot test is more detailed with many useful outputs. Thus the main concentration is scenario 2. This test was made without QoS mechanisms to substantiate the performance of the UMTS network in the way it is represented today.

In the chariot test we tried to simulate a conversation between two users communicating with VoIP. We decided that the output parameters were sufficient to evaluate a network such as the UMTS network. The test is split in two parts because of simplex measurements, one for the uplink (64 kbps) and one for the downlink (384 kbps).

The testing environment had the capability of using different codec and voice activity, which turned out useful for our understanding and for evaluating the feasibility of VoIP calls in networks today. The uplink test results are explained thorough with detailed description on how different variables effect a voice conversation. The downlink test results are explained from the uplink test results and will only conclude the essential outcomes.



9.1 Scenario 1 – Trace Route

We used the trace Route program from the Laptop B towards two destinations, a PC at Agder University and the Universal Resource Locator (URL) www.google.com. Trace Route is a program that checks how many hops it takes for a node to contact another node on a network and it give the Round Trip Time (RTT) of the. Keep in mind that these measurements only consist of pings and do not include codec operations or other factors that would increase RTT. RTT is a measure on how long it takes to send a message from one end of a network to the other and back. By the use of the command “tracert”, we got the result as shown in Figure 9-1.

```
C:\WINNT\System32\cmd.exe
C:\>tracert 128.39.203.19

Tracing route to sgrm275.grm.hia.no [128.39.203.19]
over a maximum of 30 hops:
  0  0 ms    0 ms    0 ms    C:\>
  1  261 ms   260 ms  260 ms   client13_1.siemens.uat [172.16.13.1]
  2  250 ms   261 ms  260 ms   client1_2.siemens.uat [172.16.1.2]
  3  261 ms   250 ms  251 ms   212.169.122.33
  4  270 ms   261 ms  260 ms   212.169.96.217
  5  240 ms   261 ms  260 ms   212.45.188.201
  6  260 ms   260 ms  261 ms   212.169.122.65
  7  260 ms   260 ms  261 ms   e22-r01-gw01.osl200.netcom.no [212.45.190.83]
  8  250 ms   260 ms  261 ms   oslo-gw1.uninett.no [193.156.90.1]
  9  260 ms   260 ms  261 ms   stolav-gw.uninett.no [128.39.46.250]
 10 260 ms   260 ms  261 ms   kristiansand-gw.uninett.no [128.39.46.62]
 11 260 ms   281 ms  270 ms   grimstad-gw.uninett.no [128.39.0.245]
 12 280 ms   280 ms  261 ms   sgrm275.grm.hia.no [128.39.203.19]

Trace complete.
C:\>_
```

Figure 9-1: Trace route from Laptop to a PC located at Agder University College

The first column in the figure, the hop count, represents the number of stops the test packet has made along the route to attempt to contact the other computer. The next three columns are the round-trip times in milliseconds for three different attempts to reach the destination. The last column is the name of the host that responded to the request.

We can clearly see that the bottleneck is the first access point; `client13_1.siemens.uat [172.16.13.1]`, meaning the UMTS terminal. Already at this point we measured a RTT value of 261 ms which implies a one way delay of approximately 130 ms. If a call actually had been made to a subscriber at Agder University College we would experience a one way delay of approximately 140 ms.

To gain knowledge about the assumed delay in a voice call to a destination far away, we ran a trace route to the destination `www.google.com`, located in Santa Clara, CA, USA, see Figure 9-2.



```

C:\>tracert google.com

Tracing route to google.com [216.239.33.100]
over a maximum of 30 hops:

  0  260 ms  261 ms  260 ms  client13_1.siemens.uat [172.16.13.1]
  1  251 ms  260 ms  261 ms  client1_2.siemens.uat [172.16.1.2]
  2  250 ms  260 ms  261 ms  212.169.122.33
  3  270 ms  260 ms  261 ms  212.169.96.217
  4  260 ms  260 ms  261 ms  212.45.188.201
  5  260 ms  261 ms  260 ms  212.169.122.65
  6  260 ms  261 ms  260 ms  e22-r01-gw01.os1200.netcom.no [212.45.190.83]
  7  250 ms  261 ms  260 ms  oso-okr-i2-seria12-1.telia.net [213.248.78.45]
  8  260 ms  261 ms  260 ms  oso-okr-i1-pos1-0.telia.net [213.248.67.101]
  9  280 ms  271 ms  260 ms  s-bb1-pos0-0-0.telia.net [213.248.66.89]
 10  281 ms  280 ms  281 ms  kbn-bb1-pos0-0-0.telia.net [213.248.65.25]
 11  361 ms  360 ms  361 ms  nyk-bb1-pos1-1-0.telia.net [213.248.66.126]
 12  350 ms  351 ms  351 ms  nyk-i2-pos0-0.telia.net [213.248.82.10]
 13  370 ms  361 ms  360 ms  iar1-so-3-2-0.NewYork.cw.net [208.173.135.33]
 14  361 ms  371 ms  350 ms  agr1-loopback.NewYork.cw.net [206.24.194.101]
 15  350 ms  361 ms  350 ms  dcr1-so-6-0-0.NewYork.cw.net [206.24.207.49]
 16  440 ms  441 ms  441 ms  dcr2-loopback.SantaClara.cw.net [208.172.146.100]
 17  431 ms  440 ms  441 ms  bhr2-pos-0-0.SantaClarasc3.cw.net [208.172.156.214]
 18  451 ms  450 ms  681 ms  csr1-ve241.SantaClarasc3.cw.net [216.33.153.188]
 19  441 ms  441 ms  440 ms  google-exodus.exodus.net [64.68.64.210]
 20  451 ms  440 ms  441 ms  216.239.47.2
 21  460 ms  451 ms  461 ms  www.google.com [216.239.33.100]

Trace complete.

```

Figure 9-2: Trace route from Laptop to destination www.google.com

This trace route test had the same basis as the test described above but the latency achieved with a much further distance gave us a RTT value of 460 ms or in one way delay approximately 230 ms. Note that the latency from the gateway router to the destination is around 200 ms; less than the delay achieved from the internal UMTS network.



9.2 Scenario 2 – Part 1, using the uplink

This test concludes communication from a mobile terminal to a laptop connected to the same subnet. In this part the UMTS uplink is used for the call. Table 9-1 shows how the test is set up and protocols used.

Table 9-1: Test setup for scenario 1, part 1

Test setup	
Endpoint 1	172.16.14.254
Endpoint 2	172.16.1.12
Network Protocol	RTP
Service Quality	
Script Name	G.711a
Pair Comment	VoIP G.711a 1
Console Knows Endpoint 1	172.16.14.254
Console Protocol	TCP
Console Service Quality	n/a
Endpoint 1 Knows Endpoint 2 (Setup)	172.16.1.12
Endpoint 2 Setup Protocol	TCP



9.2.1 Throughput

Throughput is important when we operate with fixed bandwidth, especially in a network that is under development with few guarantees. We used a codec meant for a 64 kbps channel, which is the estimated bandwidth in a single UMTS channel.

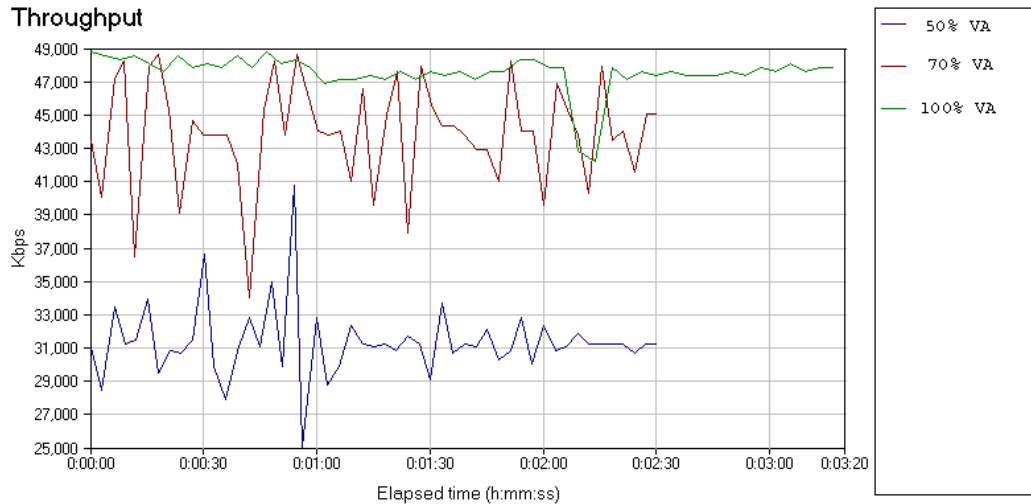


Figure 9-3: Throughput UMTS (Uplink)

To see the amount of bandwidth used with a certain voice activity, we have displayed the throughput graph from the measurements, see Figure 9-3. We simulate a call where people stops and continues to speak in a given voice activity rate. The graph presented here spans over a large range because of the voice simulation. From the figure we can observe that the greatest voice activity implies complications for the channel. Throughput is dependent on different factors easily described with a set of equations:

$$\text{Throughput} = \text{TransferSize} / \text{TransferTime}$$
$$\text{TransferTime} = \text{RTT} + 1 / \text{Bandwidth} * \text{TransferSize}$$

From Table 9-2 we can see that maximum achieved throughput is 48,841 Kbps which equals 50.01 kbps, meaning that that our estimated 64 kbps channel can deliver 50 kbps throughput which exclude the 100 % VA test in the uplink direction.

Table 9-2: Throughput results from UMTS (Uplink)

Group/pair	Average (Kbps)	Minimum (Kbps)	Maximum (Kbps)	Time(sec)
50 % VA	31,212	25,074	40,779	150,018
70 % VA	43,683	33,985	48,629	150,073
100 % VA	47,523	42,239	48,841	197,152



9.2.2 One Way Delay

In a conversation it is important to reduce the delay to the minimum. If the transmission suffers from great delay, the speakers will easily get confused during the conversation. We excluded the 100 % VA because of the poor channel capacity substantiated in Chapter 9.2.1.

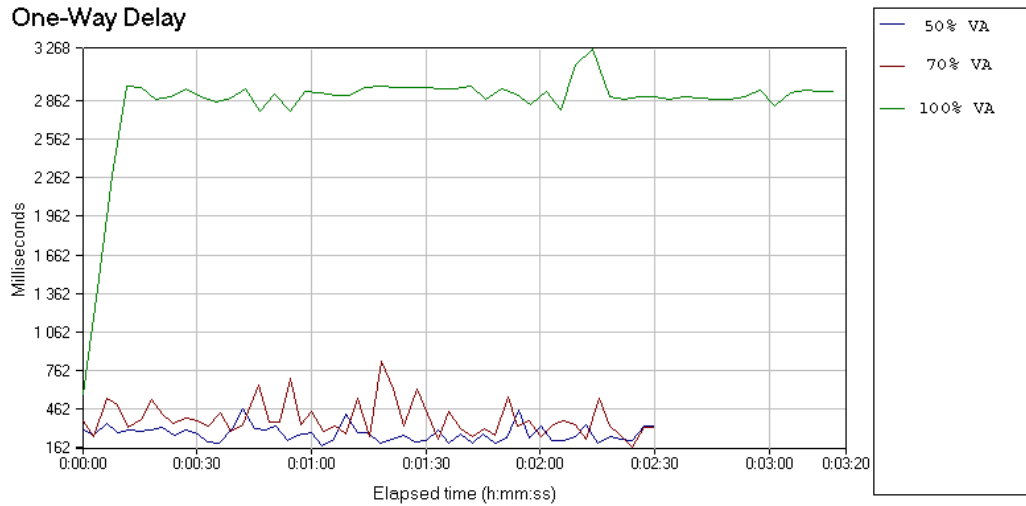


Figure 9-4: One-Way Delay measurements (Uplink)

Figure 9-4 shows the outcome of the test regarding delay. The one way delay is similar to the propagation delay; only the delay factors associated with the network (the “wire”) itself are included. Chariot measures this by synchronizing the endpoints’ timers and determining delay in a single direction. We observed that the harder we pushed the channel, the greater delay we got. The values collected from this test (see Table 9-3) gave us a hint of poor characteristics for VoIP using the UMTS uplink. If we had actually made the call with these delay-values, we would probably not be satisfied.

The results presented for 100 % VA in Table 9-3 confirm the fact that this measurement has to be excluded for further evaluation.

Table 9-3: Results from One-Way Delay measurements (Uplink)

Group/pair	End-to-End Delay Average (ms)	One-Way Delay Average (ms)
50 % VA	312	271
70 % VA	429,84	389
100 % VA	2 870,6	2 830



9.2.3 Jitter

When a datagram is sent, the sender gives it a timestamp. When it is received, the receiver adds another timestamp. These two timestamps are used to calculate the transit time of a datagram. If the transit times for datagrams within the same test are different, the test contains jitter (see Figure 9-5). In a video application, it manifests itself as a flickering image, while in a telephone call, its effect may be similar to the effect of packet loss; some words may be missing or garbled.

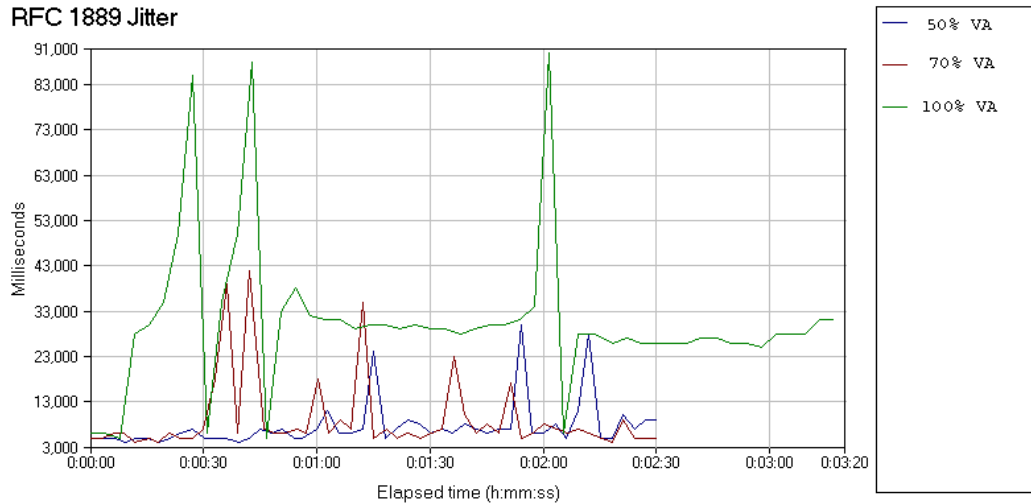


Figure 9-5: RFC 1889 Jitter measurements (Uplink)

As mentioned, jitter is damaging in both real time video and conversational. The results given in Table 9-4 is not convincing in relation to the expected jitter value of <1 ms presented in Table 6-1. The average jitter for the most common VA level (50 %) is 7,580 ms and will most likely damage the conversation.

Datagrams may be lost due to jitter buffer overruns, or datagrams that had a delay variation greater than the jitter buffer size. Overruns include datagrams that were delayed too long for the jitter buffer you configured; for example, a datagram with a delay of 50 ms would not be contained in a jitter buffer set to 40 ms. In this test we can see that the value for lost datagrams varies from 12 to 16 datagrams, meaning that there are packets which are discarded during transmission.

Table 9-4: Results from RFC 1889 measurements (Uplink)

Group/pair	RFC 1889 Jitter Average (ms)	Jitter Buffer Lost Datagrams
50 % VA	7,580	12
70 % VA	9,080	16
100 % VA	30,740	191



9.2.4 MOS Estimate

Chariot calculates a MOS, earlier described, which gives us an idea about how good a voice quality is in a conversation. This addition can give us information about how a human speaker experiences the use of VoIP. The graph in Figure 9-6 varies in the same way the coded speech does.

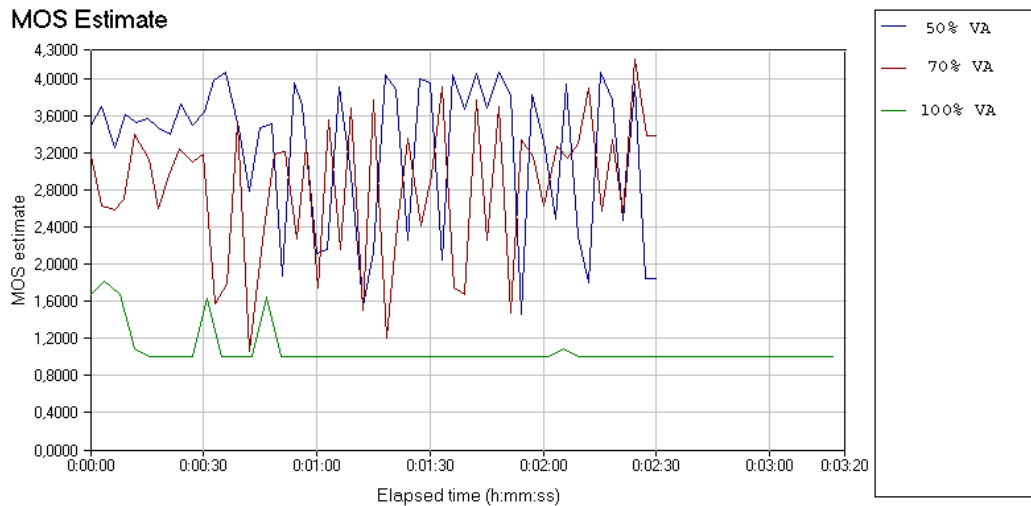


Figure 9-6: MOS Estimate measurements (Uplink)

From Figure 9-6 we can see that only the session where 50 % VA is used gives a satisfied average MOS value of 3.26, but some of the customers may even at this point become dissatisfied. If we raise the VA level up to 70 %, many or nearly all customers would feel dissatisfied with an average MOS value of 2.82.

We think of the average values as the most important ones since this indicates what the parties may think of the quality achieved from the whole conversation. The maximum/minimum –values show how poor the conversation quality can be, and in this case where the lowest value is 1.06, very poor.

Table 9-5: Results from MOS Estimate measurements

Group/ Pair	MOS Average	MOS Minimum	MOS Maximum
50 % VA	3,26	1,46	4,07
70 % VA	2,82	1,06	4,21
100 % VA	1,07	1,00	1,82



9.3 Scenario 2, part 2, using the downlink

Communication from a laptop to a mobile terminal connected to the same subnet. In this part the UMTS downlink is used for the call.

Table 9-6: Test setup for scenario 1, part 2

Test setup	
Endpoint 1	172.16.1.12
Endpoint 2	172.16.14.254
Network Protocol	RTP
Service Quality	
Script Name	G.711a
Pair Comment	VoIP G.711a 1
Console Knows Endpoint 1	172.16.1.12
Console Protocol	TCP
Console Service Quality	n/a
Endpoint 1 Knows Endpoint 2 (Setup)	172.16.14.254
Endpoint 2 Setup Protocol	TCP



9.3.1 Throughput

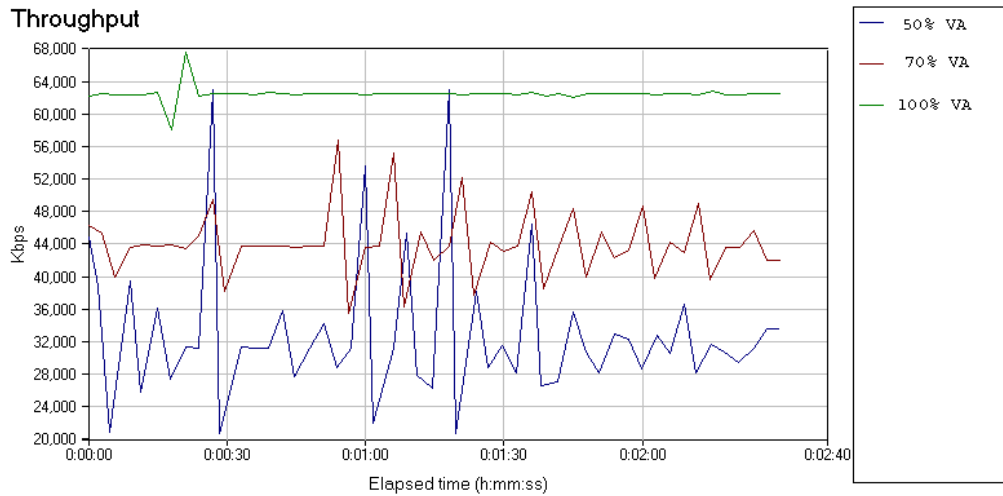


Figure 9-7: Throughput measurements (Downlink)

The downlink for UMTS has a greater bandwidth than the uplink, and we discovered that we could use all sessions; 50, 70 and 100 % VA for further evaluation. We can easily see from Figure 9-7 and Table 9-7 that the channel has a greater throughput than the uplink.

Table 9-7: Results from Throughput measurements (Downlink)

Group/pair	Average (Kbps)	Minimum (Kbps)	Maximum (Kbps)	Time(sec)
50 % VA	31,275	20,741	63,046	149,857
70 % VA	43,729	35,463	56,843	150,071
100 % VA	62,458	58,140	67,568	150,080



9.3.2 One-Way Delay

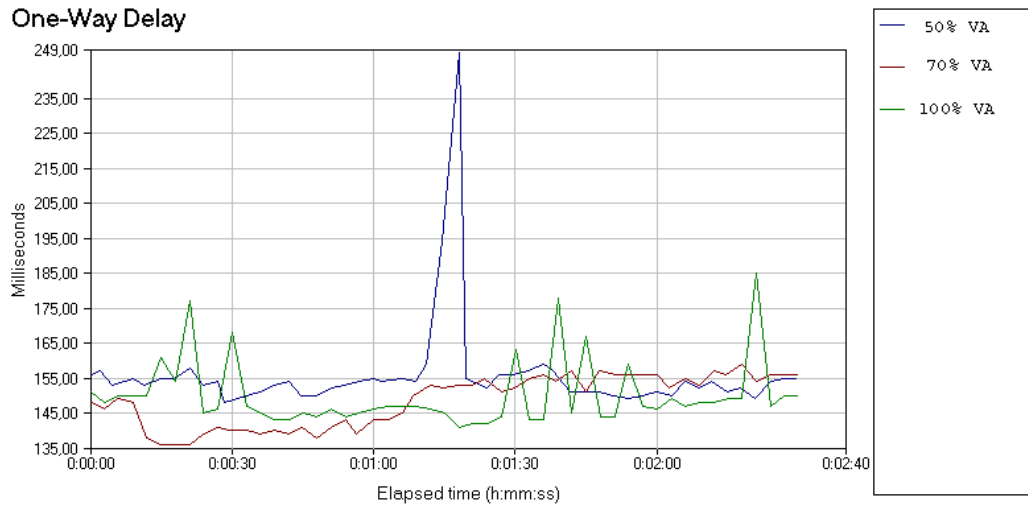


Figure 9-8: One-Way Delay measurements (Downlink)

From Figure 9-8 we found out that the delay values differed from the measurements done for the uplink, in a positive way. Note that the peak measured for the 50 % VA session is an exception from the test due to an unknown transmission failure. This may have affected the average value since we had expected the values for the 50 % VA session to be lower than the others. Table 9-8 shows exactly the delay values achieved from this test, and even though we experienced a transmission failure, these strengthened our point of view in concern of VoIP in UMTS.

Table 9-8: Results from One-Way Delay measurements (Downlink)

Group/pair	End-to-End Delay Average (ms)	One-Way Delay Average (ms)
50 % VA	196,980	156
70 % VA	189,380	148
100 % VA	191,000	150



9.3.3 Jitter

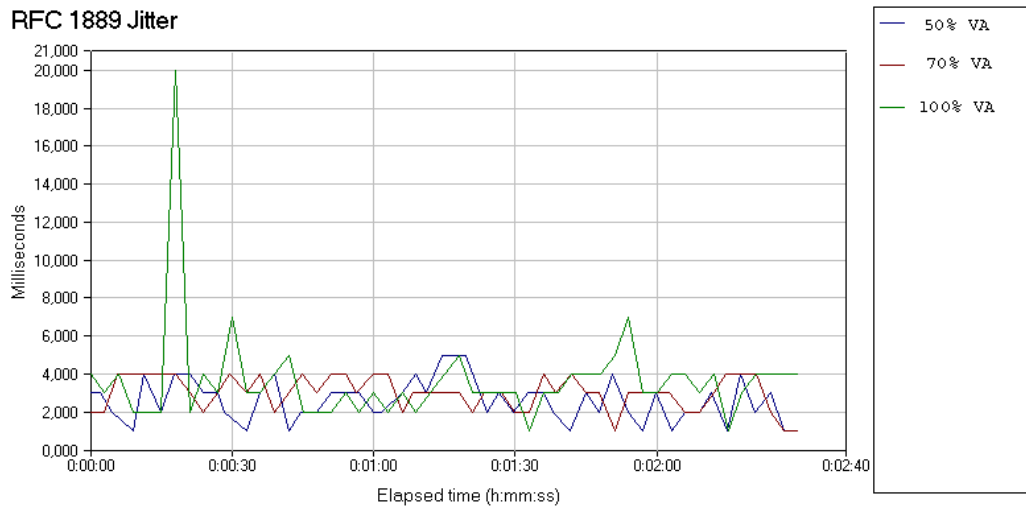


Figure 9-9: RFC 1889 Jitter (Downlink)

Delay values given in Figure 9-9 and Table 9-10 told us that the latency variation decreases when the bandwidth increase but still, the jitter-value is larger than hoped. Few datagrams are lost but the future expected goal for jitter <1ms cannot conclude satisfaction for this test.

Table 9-9: Results from RFC 1889 Jitter measurements (Downlink)

Group/pair	RFC 1889 Jitter Average (ms)	Jitter Buffer Lost Datagrams
50 % VA	2,640	2
70 % VA	3,040	0
100 % VA	3,620	8



9.3.4 MOS Estimate

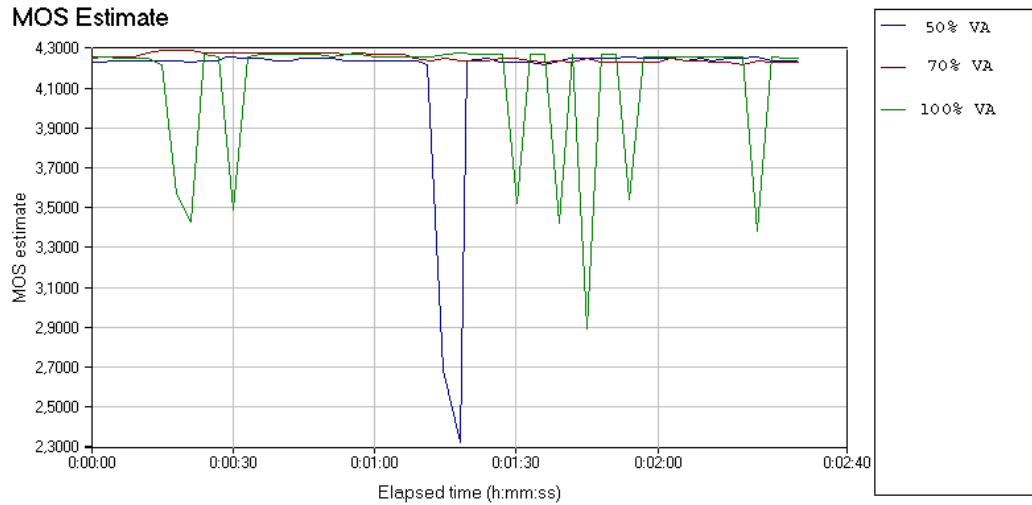


Figure 9-10: MOS Estimate (Downlink)

The MOS values given in Figure 9-10 and Table 9-10 reflect a nearly perfect conversation quality when comparing the results with Table 8-1. All sessions meet the expectations seen from the user's point of view.

Table 9-10: Results from MOS Estimate measurements (Downlink)

Group/ Pair	MOS Average	MOS Minimum	MOS Maximum
50 % VA	4,17	2,32	4,26
70 % VA	4,26	4,22	4,29
100 % VA	4,13	2,89	4,28



10 Discussion

When we started to write down the scope of this thesis, the intension was to give solutions to the use of VoIP in UMTS, cooperation with other IP network. In this chapter we will discuss the problems concerning a reliable end-to-end VoIP session. The scenarios described in Chapter 2.2 provided the basis for which VoIP session to be studied.

We start with a preliminary discussion. Then we turn our attention to the voice handling in the UMTS network. Further we discuss the QoS in relation to the scenarios. Then we will discuss the theory with the results we have got by testing within the UMTS network.

10.1 Preliminary discussion

This thesis could have been done using different approaches. We would like to focus on the choice of QoS management procedures and limitations during the testing phase.

The test did not proceed like we intended. As described earlier, there were no possibilities for setting up a TCP connection in scenario 1. The UMTS network at NetCom had restrictions in form of a firewall. Thus we had to use a trace route program to measure the latency for a long distance transmission. The results from scenario 2 were spilt into two parts, one using the downlink and one using the uplink. We decided it would be of interest to compare characteristics for the up- and down link; 64 / 384 kbps. In addition, the tests were performed with three different levels of voice activity to determine if the characteristics changed with different loads of data; 50, 70 and 100 percent VA.

More knowledge about codecs in an earlier stage could have helped us get more accurate results. In the early phase of the thesis we were not able to get information about the data rate transfer when a VoIP was traversing trough the UMTS network. The recommended codec, G.711 A-law, from Siemens, had a different data rate than the optimal AMR codec planed to use in UMTS release 5. The G.711 codec need a data rate of 64 kbps, while AMR codec have a range of 4.75 to 12.2 kbps, and the AMR-WB codec have a range of 6.6 to 23.85 kbps.

We made test files both with and without QoS management, but were not able to test QoS management due to network restrictions. This means that the discussion and conclusion concerning these technologies is based on theory only. However, we have studied the specifications and we found them promising for future network development.



10.2 Voice treatment in UMTS elements

We have seen the importance to handle the voice in a good way and as real-time traffic. The overall goal in UMTS is to present a clear voice at the right time to the receiver. Here we will discuss some concerns in a voice call that easily can be taken care of by the endpoints, and a few nodes in the network. The reason to use the chosen transport protocols will also be discussed.

By the use of up-to-date DSPs and codecs in the UMTS terminals and some network nodes (e.g. IMS-MGW) gives basis for sufficient voice quality. On the market today there are many different codec technologies supporting voice, such as the AMR codec, chosen by the 3GPP for use in UMTS. The AMR codec has the ability to adjust the data rate according to the bandwidth available. This is an important feature in a mobile network with limited coverage and bandwidth in the air interface. The voice will be affected by such adjustments.

The echo that occurs when the round-trip delay becomes greater than 50 ms, can be taken care of by an echo canceller. In the UMTS network, the echo cancelling can be taken care of by the IMS-MGW in the IMS domain.

The protocols developed for real-time traffic contributes to a reliable voice transfer described in chapter 4.4. The RTP/UDP/IP transfer protocol solution has been chosen by the 3GPP. The protocols complement each other to support VoIP, and other alternatives are not that good. For example the TCP is not invoked for voice operations because its properties can create excessive delay of the traffic. TCP has retransmission capabilities, which do not work with real-time traffic. UDP have no delaying services that could interfere with voice traffic.



10.3 QoS

In this section we discuss approaches to preserve an end-to-end QoS for VoIP sessions between two UMTS terminals and between a UMTS terminal and a terminal on the Internet. As we have defined earlier the most vital parameters for such sessions are latency, throughput, jitter and packet loss.

10.3.1 Scenario 1

An end-to-end transfer between a UMTS terminal and a terminal on the Internet is a complex matter due to the different QoS management respectively in the UMTS core network and external IP networks.

UMTS networks have the ability to co-operate with external IP networks. The GGSN take care of QoS parameter transformation from UMTS QoS to IP QoS. By default, Internet consists of routers operating on best effort basis, meaning no specified quality.

In this discussion we presume that a mobile user wants to initiate a voice call with a subscriber located in an external IP network. In concern of the mobile user and the network operator, there are normally two main expectations;

- 1) Signalling is available to prevent sending data towards a user without the capacity or the will to accept the call.
- 2) Quality of Service is available to provide a fluent conversation.

To achieve these requirements there have to be a set of functions available for QoS including forwarding treatment. IntServ or DiffServ.

- **IntServ**

IntServ uses signalling to make resource reservations. The reservations are maintained active in the routers during the whole conversation. The soft state tables in the routers could become very large due to storage of all flows going through it, resulting in high inquire time values; making IntServ scale poorly in large networks.

IntServ RSVP sends a query to the destination host, making sure of that the other party would like to initiate a conversation session. If it is a positive response the RESV message from the destination will confirm that a certain bandwidth is acceptable and the routers along the path store this information in their soft state tables. This quality is important for a conversation since we get feedback whether a reservation is successful or not. In addition the subscriber making the call will know if the subscriber at the other end has the ability to accept the call.

IntServ's poor scaling can be solved by using a combination of IntServ and DiffServ, this is still under development. IntServ could then be used in small edge networks working together with DiffServ in large backbone networks.



- **DiffServ**

DiffServ does not use signalling nor soft-state tables. DiffServ give treatment on packet-by-packet basis. Diffserv eliminates the need for per-flow state and per-flow processing and therefore scales well to large networks. Thus the benefit of Diffserv is its scalability.

The disadvantage with no signalling is that we never know if the destination is capable of receiving data with the suggested traffic class, and if the treatment is maintained when packets transit other DiffServ domains. If packets marked with a specific traffic class shall maintain the same treatment passing through transit domains, agreements between DiffServ domains are needed.

The forwarding treatment is set by the boundary routers. IP packets have to be marked with a DiffServ Code Point Field (DSCP); IPv4 TOS field and IPv6 Traffic Class. TOS is an old way of prioritizing traffic. Old routers may misinterpret DSCP field in favour of TOS, resulting in mistreatment of packets.

Both IntServ and DiffServ have specific service classes providing a fluent conversation; IntServ Guaranteed Service and DiffServ Expedited Forwarding. The operators have to use ISPs that support QoS management such as IntServ and DiffServ or both. IntServ and DiffServ are described respectively in chapter 5.2 and 5.3. If we where to promise quality end-to-end, operators have to depend on contracts with the ISPs in charge of the routers lying between a UMTS terminal and a terminal located in an external IP network.

10.3.2 Scenario 2

UMTS is moving towards an all IP structure, and it is important to give quality to different IP traffic. The quality of service is divided into 4 classes. The voice traffic is characterised as a real-time service, and is classified under the conversational class. The management for QoS in UMTS is done by the SIP sessions called QoS authorisation and resource reservation, from chapter 6.3. This indicate that the QoS in the UMTS network is considered.

There are not many problems concerning a pure UMTS VoIP transfer. Some of the reasons to this is probably because one company like Siemens or Lucent have the ability to manufacture an almost completely system, and all the elements are made according to defined rules by the 3GPP. The problems arise when the there are no resources available; the air interface is the most vulnerable point in a mobile network. If the population of subscribers in a cell increases, the bandwidth to the users can be affect and lead to more delay in the channel. The parameters concerning a VoIP call in a UMTS network have been tested and discussed in accordance with theoretical values, see chapter 0.



10.4 Tests

Both end-points were located in the same subnet, making the delay minimum compared to sessions initiated over a long distance. The estimated bandwidth, respectively 64 and 384 kbps was not honest values according to measured values. The uplink channel could not deliver enough throughput when maximum voice activity was used. A FTP session showed that the estimated bandwidth in the downlink could not be delivered. This FTP session is not documented in this thesis due to problems concerning monitor outputs and can then only be regarded as a hint.

The 64 kbps uplink average One-Way Delay varied from 271 ms for 50 % VA to 389 ms for 70 % VA. This value range is not convincing neither for a voice session between two nodes in the same subnet or through fixed networks. By comparing the MOS test-values with the estimated values presented in Table 8-1, we saw that the results were not satisfied.

3GPP requirements for one-way delay below or equal to 150 ms was partly fulfilled with the 384 kbps downlink with average measurements fluctuating from 156 ms for 50 % VA to 150 ms for 100 %. The MOS measurements reflected a satisfied voice session. The bandwidth did not cause large values of delay.

By comparing uplink- and downlink results proved the affection of bandwidth concerning delay and MOS values. The use of another codec, for example the AMR suggested by 3GPP, could improve the results for the uplink. The bandwidth needed from a 100 % VA voice using the G.711 codec is 64 kbps. By switching to the AMR codec, bandwidth down to 4.75 kbps can be a reality.

When the distance is greater, queuing and processing will play a big role for the final delay. Trace route tests from scenario 1 showed the difference between delay achieved in the UMTS network and in public IP networks, where 260 out of 460 ms was caused by the UMTS network. The tests were performed in an experimental UMTS network and there are possibilities for improvements when commercial networks are released.



11 Conclusion

This thesis presents a feasibility study of VoIP in UMTS rel. 5 interworking with external fixed networks. The feasibility study is performed through theoretical studies and practical tests. Results obtained indicate that the UMTS network is capable of offering a VoIP call.

The delay requirements from 3GPP were partly fulfilled when performing practical tests within the UMTS network. We tested with a codec requiring a bandwidth of 64 kbps which worked well in the downlink, but poorly in the uplink. More advanced coding techniques developed especially for voice treatment will require less bandwidth, as low as 4.75 kbps. This could enhance the results from the uplink measurements since the increase of bandwidth resulted in lower delay, better MOS and less packet loss. AMR is an example for such a codec and essential for a successful voice conversation in the UMTS. The AMR demands well developed DSP's for providing good speech quality.

As stated above, the bandwidth was found crucial for a successful VoIP session. Many participants and subscribers share a certain amount of bandwidth, but some traffic is more important than other. 3GPP has already specified QoS for the UMTS core network, but provision of QoS end-to-end is needed, especially when interworking with external IP networks. The GGSN acts as a gateway for external traffic and will be a central node in negotiating QoS between end-users. It is essential that the GGSN has QoS management features which support standardized QoS procedures like IntServ and DiffServ.

IntServ provides explicit feedback and strict resource reservation. Signalling with feedback gives information about the receiver's ability to accept a call. The problem with IntServ is scaling. To take advantage of the good properties supported by IntServ, the core network should implement signalling in addition to DiffServ. Scalability and signalling end-to-end could be provided with a combination of both IntServ and DiffServ. The GGSN will act as an edge router, mapping desired UMTS QoS to a behaviour aggregates for DiffServ management. In addition, the GGSN will know the end-users needs through explicit signalling to improve bandwidth utilization.

The realization of VoIP interworking with external networks depends on the implementation of QoS in Internet backbones. The network operators must eventually negotiate contracts with central ISP's to provide a reliable QoS. The ISP's must supply their routers handling voice packets with DiffServ and IntServ mechanisms. The traditional handling of best effort is not enough, even though the today's Internet is well developed.

The evaluation of QoS parameters relevant for voice quality is considered in development of standardized QoS management procedures. Development of transport, control and signalling protocols are important for a reliable voice call.



References

- [1] ITU Sub-Regional Seminar on IMT-2000, Warsaw, 2-4 October 2001, http://www.itu.int/ITU-D/tech/imt-2000/warsaw/pdf/1_3_Grillo.pdf
- [2] No. 11, Report from the UMTS-Forum, Enabling UMTS/ Third Generation Services and Applications, www.umts-forum.org, October 2000
- [3] 3GPP 3rd Generation Project Partnership, <http://www.3gpp.org>
- [4] IETF The Internet Engineering Task Force, <http://www.ietf.org>
- [5] Black, Uyles D, Voice over IP 2nd ed., Prentice Hall, 2002
- [6] S. Casner, R. Frederick and V. Jacobson, RTP: A Transport Protocol for Real-Time applications (RFC1889), January 1996
- [7] Brad Stinson, Narayan Parameshwar and Ramki Rajagopalan, UMTS Quality of Service (QoS) – An End-to-End View Award Solutions, Inc., July 2001
- [8] Nokia Networks, A History of Third Generation Mobile, www.umts-forum.org , March 2003
- [9] Kimmo Raatikainen, UMTS Architecture, <http://www.cosc.canterbury.ac.nz/teaching/handouts/cosc408/>, July 2002
- [10] UMTS World, Overview of The Universal Mobile Telecommunication System, <http://www.umtsworld.com/technology/overview.htm#a3>, July 2003
- [11] UMTSlink, UMTS , <http://umtslink.at>, 2003
- [12] 3 GPP, IP Multimedia Subsystem (IMS); Stage 2, (TS 23.228 V6.1.0), March 2003
- [13] 3GPP, Network architecture, (TS 23.002 V5.10.0) March 2003
- [14] VoiceAge, AMR-Wideband: a sound and clear Voice Revolution, <http://www.voiceage.com/amr/whychoosewide.asp>, 2001
- [15] Sjoberg J., M. Westerlund A. Lakaniemi, Real-Time Transport Protocol (RTP) Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs, (RFC3267), June 2002
- [16] 3GPP, Packet Switched Conversational Multimedia Applications; Default Codecs (Release 5), (3GPP TS 26.235 V5.1.0), March 2002



- [17] Narayan Parameshwar and Chris Reece, Advanced SIP Series: SIP and 3GPP, Award Solutions Inc, <http://www.awardsolutions.com> , 2000
- [18] IP Telephony, Camouflaging Advantages, October 2001
<http://www.ciol.com/content/flavour/voip/article/detail.asp?arid=22924>
- [19] Braden R., D. Clark, S. Shenker, Integrated Services in the Internet Architecture: an Overview, (RFC1633), June 1994
- [20] Braden R., Ed., L. Zhang, S. Berson, S. Herzog, S. Jamin, Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification, (RFC2205), September 1997
- [21] Wroclawski J., Specification of the Controlled-Load Network Element Service, (RFC2211), September 1997
- [22] Shenker S., C. Partridge, R. Guerin, Specification of Guaranteed Quality of Service, (RFC2212), September 1997
- [23] Nichols K., S. Blake, F. Baker, D. Black, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, (RFC2474), December 1998
- [24] Jacobson V., K. Nichols, K. Poduri, An Expedited Forwarding PHB, (RFC2598), June 1999
- [25] Heinanen J., F. Baker, W. Weiss, J. Wroclawski, Assured Forwarding PHB Group, (RFC2597), June 1999
- [26] 3GPP, Quality of Service (QoS) concept and architecture (Release 5), (TS 23.107 V5.8.0), March 2003
- [27] 3GPP, End to end Quality of Service (QoS) signalling flows, (TS 29.208 V5.3.1) April 2003
- [28] Yuan-Kai Chen and Yi-Bing Lin, Design of IP Connection for Gateway GPRS Support Node, <http://pcs.csie.nctu.edu.tw/DesignofIPConnectionforGGSN.ps> , May 2003
- [29] Bernet Y., P. Ford, R. Yavatkar, F. Baker, L. Zhang, M. Speer, R. Braden, B. Davie, J. Wroclawski, E. Felstaine A Framework for Integrated Services Operation over Diffserv Networks, (RFC2998), November 2000



Abbreviations

3G	Third Generations
3GPP	Third Generation Project Partner
AAA	Authentication, Authorization and Accounting
AMR	Adaptive Multi-Rate
AMR-WB	Adaptive Multi-Rate Wideband
AF	Assured Forwarding
AS	Application Server
ATM	Asynchronous transfer mode
AUC	Authentication Centre
BA	Behaviour Aggregate
CFQ	Clocked Fair Queuing
CMR	Code Mode Request
CN	Core Network
CNAME	Canonical NAME
COPS	Common Open Policy Service protocol
CS	Circuite Switching
CSCF	Call State Control Function
DEC	COPS DECision message
DHCP	Dynamic Host Configuration Server
DiffServ	Differentiated Services
DRQ	COPS Delete ReQuest state message
DS Domain	Differentiated Services Domain
DSCP	DiffServ Code Point
DSP	Digital Signal processor
EF	Expedited Forwarding
EIR	Equipment Identity Register
ETSI	European Telecommunication Standard Institute
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
GTP	GPRS Tunnelling Protocol
HLR	Home Location Register
HSS	Home Subscriber Server
I-CSCF	Interrogating Call State Control Function
ICT	Information and Communication Technology
IETF	Internet Engineering Task Force
IntServ	Integrated Services
IMS	IP Multimedia CN Subsystem
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISP	Internet Service Provider
ISDN	Integrated Service Digital Network



kbps	kilo (1000)bits per second
Kbps	Kilo (1024) bits per second
LAN	Local Area Network
MAP	Mobile Application Part
MGW	Media Gateway Function
MGCP	Media Gateway Control Protocol
MHz	Mega Hertz
MOS	Mean Opinion Score
MRF	Multimedia Resource Function
MRFC	Multimedia Resource Function Controller
MRFP	Multimedia Resource Function Processor
MT	Mobile Terminal
P-CSCF	Proxy Call State Control Function
PDF	Policy Decision Function
PHB	Per Hop Behaviours
PLMN	Public Land Mobile Network
PS	Packet Switching
PSTN	Public Switched Telephone Network
QoS	Quality of service
QoS Class	Quality of Service Class
R-SGW	Roaming Signalling gateway
RAM	Read Access Memory
RED	Random Early Detection
REQ	REQuest message, COPS
RFC	Request For Comments
RNC	Radio Network Controller
RPT	RePorT state message, COPS
RSVP	Resource reSerVation Protocol
RTCP	Real-Time Control Protocol
RTP	Real-time Traffic Protocol
RTT	Round Trip Time
S-CSCF	Serving Call State Control Function
SBM	Subnet Bandwidth Manager
SBLP	Service Based Local Policy
SDP	Session Description Protocol
SGSN	Service GPRS Support Node
SGW	Signalling gateway function
SIP	Session Initiation Protocol
SMS	Short Message Service
T-SGW	Transport Signalling gateway
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TE	Terminal Equipment
TFT	Traffic Flow Template
TOS	Type Of Service
TS	Technical Specifications



UA	User Agent
UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunication System
UNI	User network interface
URL	Universal Resource Locator
UTRAN	UMTS Terrestrial Radio Access Network
UTRA	UMTS Terrestrial Radio Access
VA	Voice Activity
VAD	Voice Activity Detection
VoIP	Voice over IP
WFQ	Weighted Fair Queuing
W-CDMA	Wideband Code Division Multiple Access



Appendixes

Appendix A – IMS Solutions from Siemens (5 pages)

Appendix B – Chariot User Guide (18 pages)

Appendix A

IMS solutions from Siemens

Appendix A

IMS solution from Siemens

The IMS system from Siemens is called IMS@vantage. With the IMS they can earn from subscribers as well as charging ASPs a fee for service delivery. IMS@vantage works together with any packet-based access network, be that UMTS packet-switched domain, GPRS, EGPRS, WLAN, or even the Internet/Intranet. This new system, part of Siemens' Internet-on-Air strategy, is based on the Session Initiation Protocol (SIP) and UMTS Release 5.

The Siemens Information and Communication Mobile Group offers the complete range of mobile solutions including mobile devices, infrastructure and applications. Devices include mobile phones, wireless modules, mobile organizers and cordless phones as well as products for wireless home networks. The infrastructure portfolio includes GSM, GPRS and 3G mobile network technologies from base stations and switching systems to intelligent networks, e.g. for prepaid services. Mobile Applications cover end-to-end solutions for Messaging, Location Based Services or Mobile Payment.

The first step is already taken: the IMS Experimental System (ES) is available and working as IMS Version 1.0. The IMS ES is a lab version of the future-commercial IMS and the first release of the IMS@vantage product line. It consists of a comprehensive infrastructure for application/service development and experimentation. IMS v2.0 is the first commercial functional IMS system for non-real-time services, and is planned to be available in May 03. Succeeding versions 3.0 and 4.0 will follow in regular hops over the next few years.

IMS v2.0 from Siemens

Siemens IMS v2.0 is at first planned to work together with the existing GPRS. The integration of IMS has to be done with as little changes and requirements to the operating system as possible. The mobile network operator (MNO) should have the possibility to append and remove the IMS system fast and without any influence to the rest of the network. The whole IMS has to work as add on to the standard GPRS. IMS must also work with UMTS Rel3/Rel4 because the operators start with UMTS soon and their investment to the GPRS IMS must be guaranteed. . This service is build only for post-paid solution. Pre-paid is not possible because it is not a real-time service.

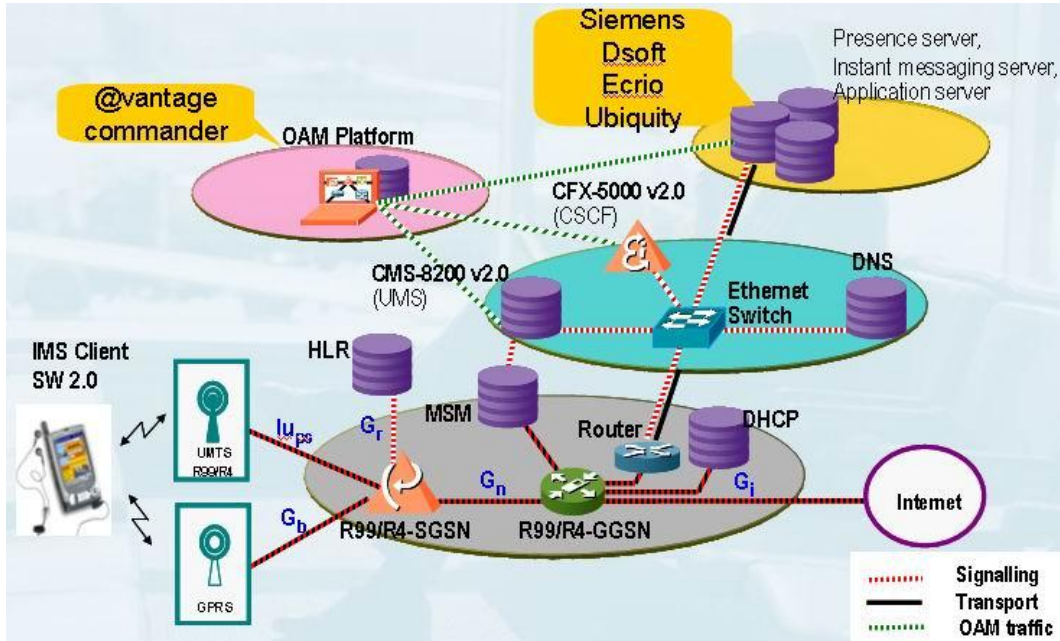


Figure 1: Siemens IMS v2.0 network architecture

The main nodes supplementing the v2.0 are CSCF called CFX-5000 and UMS (User Mobility Server) called CMS-8200, as we can see from the figure 1.

The CFX-5000 is the central network element for multimedia control. The element is divided into 3 parts as the standardised CSCF.

- Setup, maintenance, and release of SIP sessions; triggering of services

The elements features are:

- Network features
 - Time synchronisation
 - Auto registration
 - Resolution for domain names and OAM (administration, and maintenance)
- Basic features
 - Support of SIP registration and User Profile download
 - Support of authentication
 - Support of session control
 - Control of the attached SIP Application server
 - IMS service authorisation
 - Support of post-paid charging
- Services related features
 - Support of 2-party text chat
 - Support of instant messaging
 - Support of presence service

The Core Mobility server CMS-8200 acts as the Home Subscriber Server (HSS) and UMS for IMS subscriber information.

- Core mobility server, which acts as a UMTS mobility server according to 3GPP Rel 5; contains user-related information such as AAA, DNS, and Radius. It regroups the functionalities of the HLR with IMS-specific extensions.

The key functions for the element :

- Central database for network services and user profile data
- Authentication and authorization information of mobile subscribers
- User Identities
 - User identification MSISDN
 - IMSI AS Trigger
- Registration Information
 - Single sign-on
 - Registration status
 - S-CSCF SIP address
- Mobility management information
- Information for charging
 - Charging ID

IMS GPRS is a first step towards 3GPP, but it is not compliant with it. It is not real-time service, and can not provide a VoIP service. The IMS v3.0 supports real-time services, and is more of interest to this thesis.

IMS v3.0 from Siemens

IMS v3.0 is built on the v2.0, and all the nodes in v2.0 are in use. The services supported by the V3.0 are:

- Real-time Voice and video
- PSTN/CS Internetworking for voice
- Post-paid charging extensions
- Signalling based announcements
- IM-MMS Internetworking
- IPsec Tunnelling gateway
- High availability support

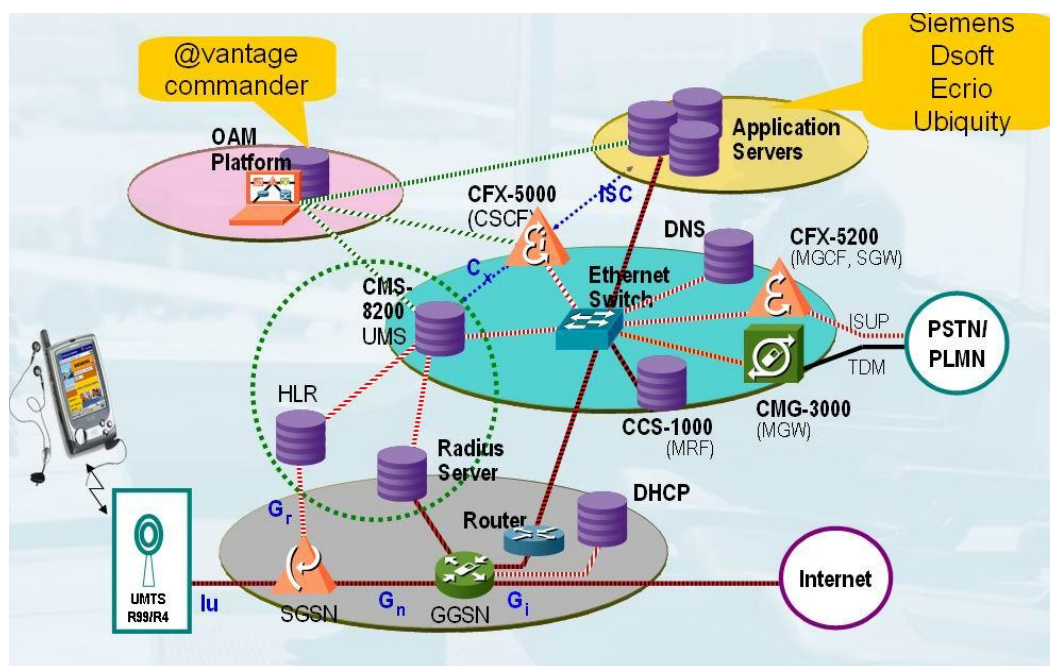


Figure 2: Siemens IMS v3.0 network architecture.

In figure 2 we can observe which new elements Siemens has decided to use for cover of IMS v3.0. MGCF called CFX-5200, MGW called CMG-3000 and MRF called CCS-3000 are the main elements in v3.0.

The Media Gateway Controller CFX-5200 performs the signalling conversation between IMS and PSTN. It is also available with an integrated Signalling Gateway (SGW).

- Interworking from SS7 to SIP

The Media Gateway CMG-3000 performs bearer conversation between the IMS and the PSTN/CS domain.

- VoIP Gateway controlled by CFX-5200

The CCS-1000 provides the multimedia resources function for media processing and conference control.

- Conference server controlled by the CFX-5000 (control and processing)

Appendix B

Chariot User guide

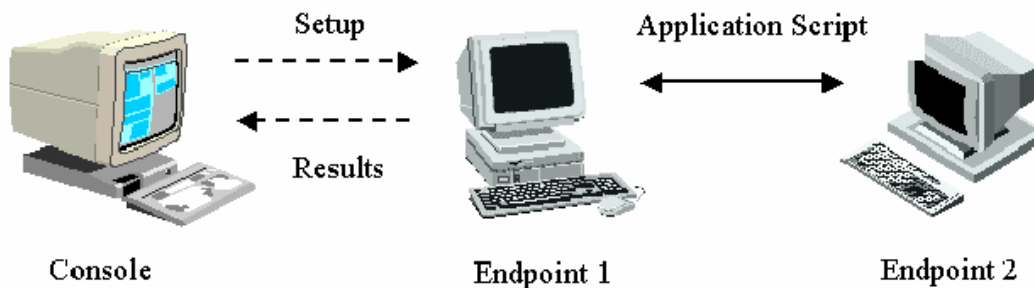
Chariot User Guide

Chariot

March 31, 2002

Getting Started with Chariot

Chariot tests and measures performance between pairs of networked computers. Using flows of real data, Chariot emulates different kinds of distributed applications, and captures and analyzes the resulting performance data. Sample Chariot tests are available at our Web site at www.netiq.com/support/chariot. Download them and try them out as you get started running your own tests.



You operate Chariot from its *Console*, a program with a graphical user interface that lets you create and run *tests*. To create a test, you determine which computers on your network to use and the type of data traffic you want to run between them. Because each computer used in Chariot tests must have NetIQ Performance Endpoint software installed, Chariot refers to each of these computers as an *endpoint*. Endpoints collect performance statistics and send them to the Console, which produces reports reflecting the response time, transaction rate, connectivity, and throughput in your network.

An *endpoint pair* comprises the network addresses of two paired computers, the network protocol used in tests between them, and the type of application to emulate for a particular test. Tests can include just one endpoint pair, or they can be more complex, involving hundreds or thousands of endpoint pairs using a mixture of network protocols and application flows.

For each test, you select an *application script* that emulates an application you're investigating. Endpoints use the script to create the same data flows that an application would send between the computers you are testing, without your having to install the application itself. Chariot comes with over one hundred predefined application scripts, providing standard performance benchmarks and emulating common end-user applications. Chariot's operations are centered at its Console. All test files and scripts are stored at the Console and distributed to the endpoints. The endpoint software is generally installed once and rarely touched afterward.

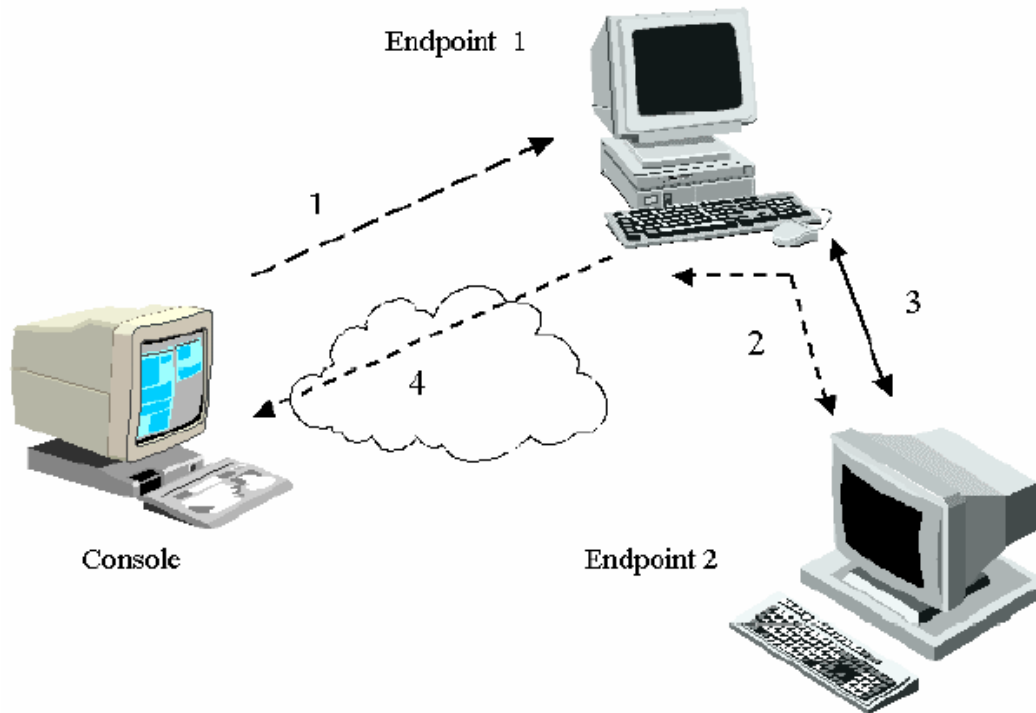
A Brief Walkthrough

Here's an example of how you can use Chariot:

1. Start the Console program from your Windows Start menu. The Console's Main window lets you create a new test or open an existing one.
2. Create a new test. A test consists of a list of endpoint pair addresses and the scripts to be sent between them. Click **New** on the File menu to define a new test.
3. Create the first endpoint pair. Specify the network addresses of Endpoints 1 and 2, and the protocol to use between them. Enter the network addresses, or select them from a list of network addresses saved from previous tests.
4. Select an application script for the endpoint pair. Chariot comes with dozens of predefined scripts that emulate common types of distributed applications. The *Application Scripts* guide lists and discusses the predefined application scripts and their settings. You can add more endpoint pairs to the test, or run it with only one endpoint pair. If you specify more than one endpoint pair, all scripts are run simultaneously.
5. Modify the variables associated with the script, if desired. A script contains variables that let you change the script's behavior. For example, in a file-transfer script, you can vary the size of the file, the number of files to transfer, and the amount of time to pause between each transfer. Data is sent and received without reading from or writing to a disk. All script variables have default values.
6. Run the test. The Console sends the scripts to each of the computers that you've specified as Endpoint 1 in your pairs. These endpoints check to ensure that they can communicate with their respective Endpoint 2 partners, and then pass them half of the selected script. When all of the endpoints are ready to start the test, the Console directs the Endpoint 1 programs to start executing the scripts. As an endpoint reaches checkpoints within a script, it creates *timing records*. These timing records are sent back to the Console, which uses them to calculate the statistics about a test run. This operation is shown in "How Chariot Works"
7. View the results. You receive a summary of the test results when a test is run. You can see the minimum, maximum, and average measurements for throughput, response time, and transaction rates. For streaming scripts, you can see how much of the data, if any, was lost. You can also see the details of each timing record created during a test.
8. Save the test. You can save test setup information and results in a file. If you want to do further processing of the results, you can export them to a .CSV file for use with a spreadsheet product like *Excel* or *Lotus 1-2-3*. You can also print results or export them in HTML format, ready to be loaded into a Web browser.

How Chariot Works

When you start a test, Chariot begins by distributing setup information from the Console to each Endpoint 1 computer when a test is run. Here's a simple example, using one endpoint pair (Endpoint 1 and Endpoint 2):



The test data flows in the above picture are labeled and described below.

1. You create a test at the Console, then click the **Run** button. The Console sends the setup information to Endpoint 1. This information includes the application script, the address of Endpoint 2, the protocol to use when connecting to Endpoint 2, the service quality to use, how long to run the test, and how to report results.
2. Endpoint 1 keeps its half of the application script, and forwards the other half to Endpoint 2. When Endpoint 2 has acknowledged that it is ready to receive data, Endpoint 1 replies to the Console. When the Console recognizes that the endpoint pair is ready (in this example, there's just one pair), the Console directs the pair to start the test.
3. The two endpoints execute their application script, with Endpoint 1 collecting the timing records that contain result data.
4. Endpoint 1 returns timing records to the Console, which interprets, analyzes, and shows the results.

NetIQ Performance Endpoints

To use Chariot, you need NetIQ Performance Endpoint software installed on each computer you plan to use in testing. NetIQ provides Performance Endpoint software free of charge. Users on your network can install endpoints on their computers by visiting our Web site at www.netiq.com/download/Endpoints and downloading the endpoint appropriate for the operating system they are using. The whole procedure should take users less than five minutes. Users should read the free PDFs available on the same Web page explaining how to install and deploy endpoints for all the operating systems supported. You can also download the endpoints and make them available for users on a shared network drive. Or you can distribute them via Microsoft's Systems Management Server (SMS). The *Performance Endpoints* guide provides more information. Endpoints for Windows support the TCP, UDP, SPX, and IPX protocols. In these environments, the endpoint runs as a service after you enable it during installation. With other operating systems, the endpoint starts automatically. It functions only during tests, and should not interfere with other application traffic on your computer. During installation of Windows endpoints, you are asked if you want to install additional endpoint support for End2End application monitoring. This support is recommended only for customers running NetIQ's End2End network monitoring product. Do not install this support if you're only using the endpoints for testing with Chariot or Qcheck. If you need to stop a Windows 95, Windows 98, Windows Me, or UNIX endpoint, go to a command prompt. From the directory where the endpoint is installed, enter

```
endpoint -k
```












To stop a Windows NT or Windows 2000 endpoint, enter











```
net stop netiqendpoint.
```

As an alternative, access the Services dialog box from the Control Panel, select **NetIQ Endpoint**, and click **Stop**.







Toolbar Icons

The icons on the toolbar in Chariot's Test window, where you'll perform most test configuration, correspond to the most commonly performed tasks. Each task is also represented in one of the Test window's 6 menus.

Creating Tests	
	Run a test
	Stop the test that is currently running
	Poll the endpoints during a running test
	Delete a pair from a test
	Add an endpoint pair to a test
	Add a VoIP pair to a test
	Add a multicast group to a test
	Edit a pair
	Replicate a pair
	Renumber pairs
	Swap E1 and E2

Viewing Endpoint Pairs	
	No grouping (default)
	Group by network protocol
	Group by script filename
	Group by Endpoint 1 address
	Group by Endpoint 2 address
	Group by Service Quality
	Group by pair group name
	Group by pair comment
	Expand all groups
	Collapse all groups

Other icons represent standard Windows tasks:

Windows Tasks	
	Save a test file
	Print the results of a test run
	Cut a pair or group of pairs from the Test window
	Copy a pair or group of pairs to the Windows clipboard
	Paste a pair from the Windows clipboard into another Chariot Test window
	Access online help for the Test window

The Main Window

After installation is complete, start the Chariot Console program by doubleclicking the Chariot Console icon in the Chariot folder. The Chariot Main window is the first window you see. Create or open as many tests as you like by choosing **New** or **Open** from the File menu. Use the Chariot Visual Test Designer to add endpoints and create tests by choosing **Design**.



Menus in the Main window include the following:

- The File Menu, used to open test files and access the Test window
- The Options Menu, used to change the default reporting, polling, and datacollection settings used by all your tests
- The Tools Menu, used to create test comparison files, customize Chariot scripts and templates, run traceroutes, and view the Error Log
- The Window menu, used to switch between the Chariot Main window and the Chariot Test window
- The Help Menu, used to access Chariot's online help files.

Each of the menus and its associated tasks is discussed in detail in separate topics

The File Menu

The File menu gives you access to the Chariot Test window, where you can configure and run tests. In the Main window, click **New** or **Open** on the File menu to open the Test window. Click **Design** to open the Chariot Visual Test Designer, a tool for creating tests. The File menu also keeps a list of recently opened and saved Chariot test files. This file list remembers up to 9 filespecs; the most recent is numbered 1, while the least recent is numbered 9. You can change the default setting for the number of recently-accessed files shown here; click **Change User Settings** on the Options menu and click the **Directories** tab.

The Options Menu

The Options menu lets you configure Chariot settings and change values that affect all your tests.

Click **Change Display Font Settings** to change the fonts used in the Test and Comparison windows. The settings you choose here do not affect the Main window, nor do they change the fonts used in exported or printed test results. Click **Change User Settings** to open a tabbed notebook of global Chariot settings you can change. Chariot's user settings affect

- Default units in which test data is recorded and displayed (**Throughput Units** tab)
- Default parameters used in all new tests you create or run (**Endpoint Pair Defaults** tab; **Run Options** tab; **Datagram** tab; **VoIP Pair Defaults** tab; **Traceroute** tab)
- Warnings you might see while setting up tests or viewing results (**Warnings** tab)
- Directories where Chariot stores your files (**Directories** tab)
- Options for printing or exporting test results (**Output** tab)
- Ranges into which data is placed when results are reported (**Result Ranges** tab)
- Options for upgrading your Chariot license or deregistering the Console (**Registration** tab)
- Parameters affecting how Chariot runs tests through firewalls in your network (**Firewall Options** tab).

Click **Help** for context-sensitive help about each tabbed notebook page in the Change User Settings notebook.

The Tools Menu

The Tools menu provides a set of tools for customizing Chariot. The following items are available in the Tools menu:

Compare Tests

Opens the Comparison window, which lets you compare the results of multiple Chariot tests.

Edit Scripts

Opens the Script Editor, which lets you modify existing scripts or create a new script. The Script Editor can also be invoked from the Test window, but if you access the Script Editor this way, your modifications are saved at the file level and are available to all new pairs associated with the script.

View Error Logs

Shows you more information on errors you receive while running tests. Provides an organized list of errors and information about them.

Edit Output Templates

Lets you specify printer output for reports ahead of time.

Edit IPX/SPX Entries

Lets you enter a series of aliases that are stored and used in Chariot tests instead of those long IPX network addresses.

Edit QoS Templates

Lets you run tests with Quality of Service (QoS) settings. Chariot now offers three options for testing with QoS or prioritized traffic:

- “Generic QoS Templates”
- “IP TOS Templates”
- “DiffServ Templates”

Run Traceroute

Lets you run a traceroute test between any two endpoints

The Help Menu

The Help menu gives you instant access to the HTML-formatted Chariot help files, which open with Microsoft Internet Explorer when selected.

Click **Contents and Index** to access the Chariot Help files for the Performance Endpoints, the Chariot API, NetIQ application scripts, and messages.

Click **Current Window** to get descriptive information about the Chariot window you are currently viewing.

Click **Shortcut Keys** for a list of all shortcut keys and key combinations available for the current window.

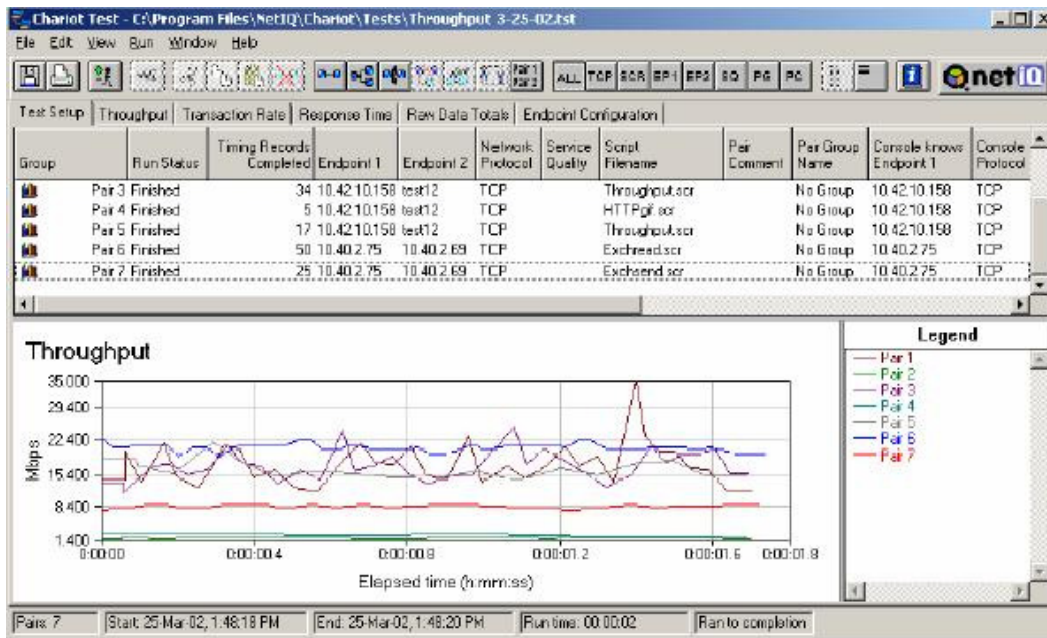
Click **About Chariot** for details on the Chariot version and build level, and for information about service and support. The About Chariot dialog box contains copyright and release information.

Click **Support Info** in the About Chariot dialog box for Chariot technical support information. The **Contents and Index** menu item gives you access to all the Chariot online help files, including a comprehensive index to Chariot Help:

- *User Guide* for Chariot
- *Performance Endpoints* Guide
- *Application Scripts* Guide
- *Message Reference*
- *API Guide* for Chariot.

The Test Window

The Test window lets you build, run, and view the results of a test. You can add new endpoint pairs, copy or delete existing endpoint pairs, edit endpoint pairs, or view the statistics and graphs of a test you've run.



Access the Test window by clicking **New** on the File menu in the Main window. Or click **Open** to open an existing test. Each endpoint pair in a test is shown as a row in the Test window. The pairs are identified by a pair number, shown on the left-hand side of each row. You can edit a pair by double-clicking it, or by selecting it and using the menus or toolbar icons. The graph icon indicates that a pair is selected for graphing. The Test window is partitioned into areas accessible by tabs. The first tab is for test setup; other tabs let you view the results of a test. You can save a test to a file, or export it in a variety of file formats. The button featuring the NetIQ logo is a hyperlink to the NetIQ Web site.

Right-click to bring up a floating Edit menu when you're pointing to any endpoint pair in test window. While in the graph region, right-click to bring up a menu that lets you change Graph Configuration or Throughput Units. Menus in the Test window include the following:

- The File Menu--to save, export, print, or clear results
- The Edit Menu--to add, change, copy, or delete pairs
- The View Menu--to change how pairs are grouped and shown
- The Run Menu--to start a test and adjust its parameters
- The Window Menu--to toggle between Chariot's windows
- The Help Menu--to get more information. This menu is identical to the

Help menus in the Main and Comparison windows. Each menu is discussed in the sections below. The **Test Setup** tab is always available; the other seven tabs are only available when a test has results. The **Throughput**, **Transaction Rate**, **Response Time**, **Lost Data**, and **Jitter** tabs display a graph of the appropriate results. Some tabs and sub-tabs may not be shown unless your Chariot license includes optional modules, or unless certain protocols are used in testing.

The File Menu

When you save a test, the test setup and results, if any, are stored together in a binary file. By default, Chariot uses the file extension `.tst` for this file. For each endpoint pair in the test, Chariot stores the names of the endpoints, their protocol and service quality, and the full script and script variables used by that pair. If results are available, Chariot saves in the binary file all the timing records associated with the most recent test run.

Save

Saves a new test or saves a test (`.tst`) file using its existing filename.

Save as

Saves a test (`.tst`) file with a new or different name. If the test only contains pairs whose functions are not supported in the Chariot version in which you are saving the test, you cannot save the test in the specified level.

Print

Prints test setup details and results. If you've entered preferences on the Output tab of the Change User Settings notebook, these are used.

Export

Exports test setup details and results to HTML, `.TXT`, or `.CSV` file format. If you've entered preferences on the Output tab of the Change User Settings notebook, these are used.

Clear Results

Erases the results from a test without affecting the test setup. Only the Test Setup tab remains in the Test window after you clear results. You might use this command to save only the setup for a test, without a large set of accompanying results that you don't plan to keep.

Close

Exits the current Test window. If a test is running in the window, you are asked if you want to stop it. If the test hasn't been saved to a file, you are asked if you want to save it.

The Edit Menu

The items in the Edit menu let you change the setup for a test. In addition to editing, deleting, and copying individual endpoint pairs, you can select multiple pairs or multicast

groups and perform these same operations across all of them. You can also cut and paste pairs among different test windows. Right-click to bring up a floating Edit menu when you're pointing to any endpoint pair in a test window. Following are brief descriptions of each item in the Edit menu.

Cut

Removes a selected pair or group of pairs from the Test window and places them on the Windows clipboard. When selecting multiple pairs, hold down the Shift key and click the first and last pairs to be cut. The two pairs you clicked, as well as all pairs in between, are selected. If the Test window contains test results, you are prompted to confirm the cut because a cut operation clears all test results. You are given the option to cancel or proceed.

Copy

Copies a selected pair or group of pairs from the Test window to the Windows clipboard. When selecting multiple pairs, hold down the Shift key and click the first and last pairs to be copied. The two pairs you clicked, as well as all pairs in between, are selected. Copying does not clear test results.

Paste

Takes a pair or group of pairs from the Windows clipboard and pastes them into the Test window. You can only paste Chariot data into the Test window. But you can paste data cut or copied from Chariot into other applications that accept tab-delimited data, such as spreadsheets and editors. Before pasting the data into the selected Test window, Chariot ensures that the Paste operation does not exceed the licensed number of endpoint pairs. If this number will be exceeded by the Paste operation, a dialog box alerts you and the paste operation is aborted. If the Test window contains test results, you are prompted to confirm the paste operation because it will clear all test results. You are given the option to cancel the paste operation or proceed.

Delete

Removes a selected pair or group of pairs from the Test window. To delete multiple pairs, hold down the Shift key and click the first and last rows to be removed. When you attempt to delete a pair, a warning asks, "Are you sure you want to delete the selected endpoint pair(s)?" You can disable this warning in the Change User Settings notebook.

Select All Pairs

Selects all of the pairs in the Test window. You must select a pair before you can cut, copy, or mark it.

Deselect All Pairs

Deselects all of the highlighted pairs in the Test window.

Mark Selected Items

Specifies a pair or pairs for inclusion in a graph or in a printed report. New pairs are initially marked when they're created. This command marks all pairs and groups that are

currently selected in the Test window. A column to the left of the pairs displays a graph icon to indicate that pairs and groups are marked.

Unmark Selected Items

Excludes a pair or group from graphs or printed reports. Unmarks all pairs and groups currently selected in the Test window.

Edit

Lets you modify a highlighted pair, set of pairs, or a multicast group in the Test window. If you highlighted multiple pairs, the Edit Multiple Endpoint Pairs dialog box opens so that you can modify the definitions of many pairs simultaneously. You cannot edit multiple multicast groups simultaneously, nor can you edit multiple multicast groups or a combination of multicast pairs and single pairs.

Edit Pair Setup

Lets you use a different network protocol or service quality for setup and results flows between the Console and Endpoint 1, or between Endpoint 1 and Endpoint 2.

Replicate

Lets you duplicate an existing pair, multicast group, or group of pairs. First highlight the pair(s) to be replicated. You must replicate pairs and multicast groups separately. In the Replicate Selected Pairs or Replicate a Multicast Group dialog box, enter the number of replications desired.

Swap Endpoint 1 and Endpoint 2

Changes the roles of the endpoints in the pair(s). First highlight the pair(s) of endpoints to be swapped. Swapping is not allowed if any of the selected pairs has a different pair setup address defined, or if a multicast pair is selected.

Add Pair

Lets you add a pair of endpoints to a Chariot test. See “Adding or Editing an Endpoint Pair”

Add Multicast Group

Lets you create a group of receivers for an IP Multicast test. See “Adding or Editing a Multicast Group” information.

Add VoIP Pair

If you’ve purchased the VoIP Test Module, lets you add a VoIP endpoint pair to a Chariot test. See “Adding or Editing a VoIP Endpoint Pair”.

Renumber All Pairs

Renumbers all the pairs in the Test window sequentially, beginning with 1. After you have edited or deleted pairs within a test window, pairs aren’t automatically renumbered.

The Run Menu

The Run menu lets you run tests and traceroutes and modify the way Chariot tests are performed.

Run

Starts a test that you've created.

Stop

Stops a running test before it completes. When you stop a running test, a warning box appears asking, "A test is currently running. Do you want to stop the test?" Click **Yes** to stop the test or click **No** to resume the running of the test.

Run Traceroute

Runs a traceroute between a pair of endpoints in a test.

Set Run Options

Sets parameters determining how one test is run. A two-page notebook is shown, with a Run Options page and a Datagram page. Set run options for all tests by clicking the Run Options tab in the Change User Settings notebook.

Poll Endpoints Now

Causes the Console to contact each of the Endpoint 1 computers in a test while a test is running. The endpoints reply, returning the number of timing records they've created so far in this test. Poll Endpoints Now corresponds to the Poll icon on the toolbar.

The View Menu

The View menu lets you alter the way information is displayed in the Test window and the Comparison window. For more information, see the separate topics for each menu item.

Sorting

You can reorganize the way pairs are shown in the Test window according to a series of different criteria. Clicking **Sort** on the View menu opens the Sort dialog box, where you specify the first, second, and third criteria by which the pairs will be sorted.

Sort by

Specifies the first sorting criterion. The list contains the parameters applicable for sorting the test data on the tab now shown in the Test window.

Ascending

Sorts the data in order from smallest to largest values, or in alphabetical order if hostnames or group names are used for the endpoints.

Descending

Sorts the data in order from largest to smallest values, or in reverse alphabetical order if hostnames or group names are used for the endpoints.

Then by

Specifies the second and third sorting criteria.

Group By

The Group by feature lets you determine how pairs are shown in the Test window. Once pairs are grouped, their results appear in these groups when you print or export. By default, the no-grouping category “All Pairs” is used. Each grouping and sorting option corresponds to an icon on the toolbar. Refer to “Toolbar Icons”

Sort By

Lets you specify whether to sort pairs in ascending or descending order.

Group Sort Order

Lets you sort the groups of pairs in either ascending or descending order.

Other View Menu Items

Information

Lets you choose which aspect of a test’s results are graphed and shown in the Test window. If a test is still being configured, only the Test Setup tab is available. While a test is running, or after a run, all tabs that contain results are shown. If the test does not contain RTP pairs, for example, the Jitter tab is not shown.

Expand All Groups

Lets you view details about all pairs in a test. Test setup and results are displayed for all of the pairs.

Collapse All Groups

Decreases the level of detail shown in the Test window; details for individual endpoint pairs are not shown. Instead, endpoint pairs are subsumed within groups. With large tests, it’s a good idea to collapse groups before you try to graph their results.

Disable Graphing

Shows test results without graphing them. This option may greatly improve GUI performance if you’re running large tests, which can be tricky to graph. Automatically selected when you run a test of more than 5,000 endpoint pairs.

Graph Configuration

Lets you control how test data is graphed.

Throughput Units

Lets you change the units in which data is displayed.

Show Error Message

Displays any error message associated with the selected pair.

Show Warning Message(s)

Displays any warning message associated with the selected pair.

Show Timing Records

For pairs that have results, provides a breakdown of results by timing record.

Show Endpoint Configuration

Provides information about the endpoint computers

The Window Menu

The Window menu lets you move among your open windows in Chariot.

Main Window

Brings Chariot's Main window to the foreground. If you are currently viewing the Main window, this menu item is not shown.

Comparison Window

Brings Chariot's Comparison window to the foreground, if you have opened a comparison. You can open as many as 9 Test windows simultaneously. Each open Test window is shown on the Window menu. Click the name of a test to bring the Test window to the foreground.

Test Status

The Status Bar, at the bottom of the Test window, shows summary information about the progress of a test as it moves from initialization to completion. The status bar can include up to five fields, depending on the test's current state. Status Bar Fields include the following:

- The number of pairs used in the current test.
- Start status (shown when the test is initializing and running)--gives the overall progress of the test. When the test ends, this field displays the start date and time of the current test;
- End status (only shown while the test is running)--indicates the elapsed time (hr:min:sec). When the test ends, this field shows the ending date and time.
- Duration (shown while the test is running)--estimates the time remaining until the test will complete (hr:min:sec). This field shows no value until enough timing records are received to calculate an estimate. When the test ends, this field shows the actual total run time. You might see the estimated remaining

time increase in large increments, if one or more pairs are using random SLEEP durations.

- Completion status (displayed when the test ends)--informs you whether the test ran to completion, was stopped by the user, or stopped because an error was detected.