ARROWS QoS Framework

Manuel Ricardo[1,2], Jaime Dias[1,3], Gustavo Carneiro[1], José Ruela[1,2]

1. INESC Porto – Instituto de Engenharia de Sistemas e Computadores do Porto

2. FEUP – Faculdade de Engenharia da Universidade do Porto

3. ISEP – Instituto Superior de Engenharia do Porto

31/8/2002

Abstract:

This document describes the QoS framework developed under the ARROWS project. This QoS framework provides a solution for deploying IP based multimedia applications over UMTS access networks and provides these applications with the Quality of Service they require. The solution assumes that IntServ/RSVP is supported in the UMTS access network that, in turn, can interoperate with DiffServ IP core networks.

Acronyms

3GPP	Third Generation Partnership Project
API	Application Programming Interface
BR	Border Router
CBQ	Class Based Queuing
DSCP	Differentiated Services Code Point
DSS	Darwin Streaming Server
ER	Edge Router
GGSN	Gateway GPRS Support Node
GSM	Global System for Mobile Communications
IETF	Internet Engineering Task Force
IP	Internet Protocol
MPEG	Motion Picture Expert Group
MS	Mobile Station
MT	Mobile Terminal
MTU	Maximum Transmission Unit
NAS	Non Access Stratum
NSAPI	Network Access Point Identifier
PDP	Packet Data Protocol
PHB	Per Hop Behaviour
QoS	Quality of Service
RAB	Radio Access Bearer
RAT	Robust Audio Tool
RRM	Radio Resource Management
RSPEC	Receiver SPECification
RSVP	Resource ReSerVation Protocol
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol
RTSP	Real-time Transport Streaming Protocol
SAP	Service Access Point
SAP	Session Announcement Protocol
SAPI	Service Access Point Interface
SDP	Session Description Protocol
SGSN	Serving GPRS Support Node
SIP	Session Initiation Protocol
TBF	Token Bucket Flow
TCP	Transmission Control Protocol
TE	Terminal Equipment
TFT	Traffic Flow Template
ToS	Type of Service
TSPEC	Traffic SPECification
UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
UTRAN	UMTS Terrestrial Radio Access Network
WCDMA	Wideband Code Division Multiple Access

TABLE OF CONTENTS

1 INTRODUCTION

This document describes the QoS framework defined under the ARROWS project. This framework provides a solution for deploying IP based multimedia applications over UMTS access networks and provides these applications with the Quality of Service they require. The solution assumes that IntServ is available at the IP access networks.

We propose to deploy IntServ/RSVP over an UMTS access network that, in turn, can interoperate with DiffServ core networks. The main problems found and solved during this work are concerned mainly with: (1) the difficulty of deploying RSVP on wireless links, (2) the asymmetry of the interfaces at TE and GGSN for activating UMTS bearer services, (3) the aggregation of IP flows into PDP Contexts, and (4) the mapping of IP QoS parameters into UMTS QoS parameters. This document is composed of 5 sections.

Section 2 describes the problems and characterises the Quality of Service in packet (read IP) switched networks. We start from the applications point of view – first classify them according to their requirements and then review the IETF multimedia stack. A simple framework is presented next, so that the problem can be delimited. Then, an overview of QoS in IP networks is given – we present the IntServ, DiffServ and IntServ over DiffServ approaches, by showing what network equipments have to do in each approach. UMTS QoS is reviewed next by describing the QoS architecture proposed by 3GPP, the main attributes of the UMTS bearer service and the policing mechanism.

Section 3 presents the proposed solution. We begin by presenting the functional architecture. Then, the key block of our architecture (the QoS manager), used in mobile terminals, is presented. Next, it is shown, by means of an example, how RSVP can live with UMTS signalling. The usage of the IP datagram ToS field to request a service to UMTS is described next. The QoS API and the strategy chosen for mapping IP QoS parameters into UMTS QoS parameters follow. After the presentation of the IP traffic control solution implemented, the aggregation of IP flows into PDP Contexts used in ARROWS, along with the selected values, is described.

Section 4 aims at proving the proposed solution. The solution described in this document has been already implemented and deployed over an emulated network. We start by describing the test architecture and the analysis tool that has been developed for observing the behaviour of the implementation. In particular, we help the reader in understanding arrival/service curves and delay plots. Finally, the results obtained for the video telephony and audio-video streaming services are presented and discussed.

Section 5 concludes the report, by reviewing the main contributions of this work.

2 QOS IN PACKET SWITCHED NETWORKS

2.1 The applications point of view

Applications can be classified with respect to QoS according to two commonly accepted taxonomies. The first comes from the IP world, and the second from 3GPP.

In the IP world, applications can be first classified as elastic or real-time. Elastic applications are those whose flows can be extended in time; these applications are sometimes further classified as interactive, interactive bulk or asynchronous. An example of the first is Web browsing, the second is FTP, and the last mail transfer. Real time applications are those for which the arrival time of a packet at the receiver is relevant - if it misses a pre-defined deadline, it is of no use for the application. Real time applications, in turn, may be classified as tolerant or intolerant, depending on whether they can tolerate or not some loss of data. Audio and video are real-time tolerant applications (audio more than video) while a robot arm controlled through the Internet is an example of an intolerant real time application.

An alternative taxonomy, used by 3GPP, classifies applications into four classes according to the characteristics of the traffic they generate: conversational, streaming, interactive and background [4]. A conversational class preserves the time relation between the information entities of the stream (the delay jitter must be tightly controlled), the end-to-end delay is required to be low and the traffic is usually almost symmetric. A typical example is audio conversation, but video telephony is likely to become very common, as well. A streaming class is used for transferring data so that it can be processed as a steady continuous flow. Streaming applications, like video streaming, are usually highly asymmetric and end-to-end delay and jitter need not to be as tightly controlled as in the conversational class. The interactive class includes applications like web browsing, where traffic is asymmetric and round trip time as well as bit error ratio should be kept low. The background class serves applications like email, where delay is not an important issue.

Providing a QoS solution means, among other aspects, supporting a transport solution capable of satisfying these classes of applications, which run on terminal equipments.

2.2 The IETF Multimedia Architecture

IETF proposes a functional architecture for supporting multimedia applications over IP networks that consists of the following protocols: SIP – Session Initiation Protocol; SAP - Session Announcement Protocol; SDP - Session Description Protocol; RTP - Real-time Transport Protocol; RTCP - Real-time Transport Control Protocol; RTSP - Real-time Transport Streaming Protocol; RSVP - ReSerVation Protocol.

SIP is an application-layer control protocol used to establish, maintain and terminate multimedia sessions or calls [16]. These multimedia sessions can be conferences or IP telephony calls. SIP can also be used to invite members to sessions previously advertised by other means, such as SAP. The mechanisms provided by SIP can be used by end systems or proxy servers that use them for instance, to set-up a call, forward a call or negotiate a terminal capability.

SAP provides mechanisms to assist the advertisement of multicast multimedia conferences or other multicast sessions [17]. To communicate the relevant session set-up information, a session directory may be used. An instance of such a session directory periodically multicasts packets containing a description of the session, and these advertisements are received by other session directories, so that potential participants can use the session description to start the necessary tools required to participate in the session.

SDP provides the means to convey such information to recipients [18]. SDP defines the format for description; it does not include a transport protocol and can use different transport protocols including SIP, SAP and RTSP.

RTP provides end-to-end delivery services of data with real-time characteristics, such as interactive audio or video streaming [19]. RTP is typically used over UDP (User Datagram Protocol) since, in real-time, timely delivery can be more important than data reliability. RTP provides the following functions:

- **Time stamping**. Used to reconstruct the original timing in order to play data with the correct rate. It is also used for the application to synchronize audio and video.
- **Sequence Numbering**. Used to place incoming packets in the correct order since UDP does not provide that for itself. It is also used for packet loss detection. When, for instance, a video frame is split into several RTP packets, all of them can have the same timestamp. Therefore, timestamp is not enough to place the packets in the right order.
- **Payload type identification**. Allows the application to know how to decode data. Default payloads are defined in RFC 1890. Examples include PCM, GSM, G.722, G.728, MPEG1/MPEG2 audio and video, H.261.
- **Source identification**. Allows the receiving application to know where data is coming from. In a videoconference it may be used to distinguish who is talking.

RTCP is the counterpart of RTP and provides the control services. The primary function of RTCP is to provide feedback on the quality of the data distribution. In an RTP session, participants periodically send RTCP packets. RTCP is typically delivered in UDP datagrams. These control packets provide the following services:

- **QoS monitoring and congestion control**. The sender can adjust its transmission rate based on receiver report feedback. Receivers can determine whether congestion is local or global.
- **Source identification**. In RTP, 32-bit random numbers identify the sources. These are not appropriate for human users. RTCP conveys textual information about the participants.
- **Inter-media synchronization**. RTCP sender reports contain an indication of real time and the corresponding RTP timestamp. This can be used in inter-media synchronization like lip synchronization in video.
- **Control information scaling**. 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 sending its control packets to all the others, each one can independently observe the number of participants. This number is used to calculate the rate at which the packets are sent.

RTSP is a client-server multimedia presentation protocol to enable controlled delivery of streamed multimedia IP data [20]. It provides "VCR-style" remote control functionality for audio and video streams, like pause, fast-forward, reverse, and absolute positioning. Sources of data include both live data feeds and stored clips. RTSP works with protocols like RTP and RSVP to provide a complete streaming service over Internet. It also provides means for choosing delivery channels (such as UDP, multicast UDP and TCP), and delivery mechanisms based upon RTP. It works for large audience multicast as well as single-viewer unicast.

RSVP is a protocol used by a host to request the IP network specific end-to-end qualities of service for a certain application data stream [6][12][24][25]. The structure and contents of the QoS parameters are documented in specifications developed by the Integrated Services

Working Group [21][22][23]. It is a control protocol and occupies the place of a transport protocol on the protocol stack, on top of IPv4 or IPv6. RSVP is implemented not only in end systems but also in routers, which use them to establish and maintain the requested service along the path(s) of the flow. RSVP requests will generally result in resources being reserved in each node along the data path. The main characteristics of this protocol can be summarized as follows:

- **Unicast and multicast**. RSVP makes reservations for both unicast and multicast transmissions. It was, in fact, designed specifically for multicast and much of its complexity results from this. Unicast is treated as a special case of multicast.
- **Simplex**. RSVP makes reservations for unidirectional data flow. Data exchanges between two end systems require separate reservations in the two directions.
- **Receiver-initiated reservation**. The receiver of a data flow initiates and maintains the resource reservation for that flow.
- **Maintaining soft state in the Internet**. RSVP uses a soft-state approach in which the reservation state is cached in intermediate routers and periodically refreshed by end systems.
- **Providing different reservation styles**. These allow RSVP users to specify how reservations for the same multicast group should be aggregated at the intermediate switches. This enables a more efficient use of Internet resources.
- **Transparent operation through non-RSVP routers**. Routers that do not implement RSVP will simply use a best-effort technique, providing transparent operation through non-supporting regions.
- **Support of IPv4 and IPv6**. RSVP can exploit the ToS field in the IPv4 header and the Flow Label field in the IPv6 header.

2.3 A simple framework

In order to understand and discuss UMTS QoS solutions in the context of all-IP networks, a simple framework must be introduced. The simplest scenario representing the QoS issues addressed, consists of the network configuration shown in Figure 2-1, having the following components:

- an IP core network supporting, at least, DiffServ;
- an IP access network based on UMTS. It is the portion of the IP network for which a QoS solution will be provided;
- a terminal equipment based on IP and having an UMTS network interface. This interface provides the 4 UMTS traffic classes described in 3GPP standards;
- a server, accessible through the IP backbone, with which the terminal communicates;
- a GGSN (Gateway GPRS Support Node), acting as an IP access router and used to interconnect the UMTS based IP access network to the IP core network.

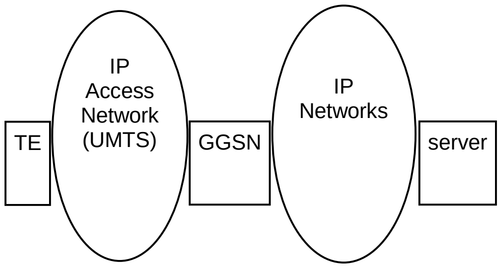


Figure 2-1 Reference Architecture

The scenario presented above assumes all services provided over IP networks. UMTS is the layer 2 technology used in the access network.

2.4 IP QoS

Current Internet transports datagrams according to a best-effort model. In practice, it means that an IP router is neither required to guarantee the delivery of a packet nor the delay for transferring it between input and output ports. Elastic applications are suited to operate in this way and, by using the congestion control mechanism of TCP, they can at each moment use the maximum bandwidth available between the source and the destination. Real time applications, however, have difficulties in living with the best effort model, since packets not delivered in time are of no use. In order to overcome this problem, IETF has defined two well-known approaches for solving QoS at IP level [11][13] – the Integrated Services (IntServ) model and the Differentiated Services (DiffServ) model. More recently, a new model enabling the use of IntServ over DiffServ domains was proposed, as well.

2.4.1 IntServ

The IntServ approach consists of 2 new classes of service, the Guaranteed and the Controlled Load, which can work both for unicast and multicast traffic. Only the Guaranteed class is of interest to our approach.

The Guaranteed service of IntServ aims at providing datagrams with a known maximum transfer delay between the source and destination hosts. To achieve this goal, signalling prior to data transfer is required and the RSVP protocol is normally used. Assume that a client application process bound to some UDP port in the terminal intends to send data to an application process bound to another UDP port in the server. If these packets require a maximum delay from the terminal to the server and the IP network elements support the Guaranteed service, resources at every IP network element along the data path will have to be negotiated and reserved for this sequence of packets, also known as a flow.

Thus, the terminal, using RSVP, issues a message (the RSVP PATH message) to the first router indicating, among other attributes, the characteristics of the flow. The first router estimates the time a packet of this flow will have to wait in queue plus the time it takes to be transmitted (plus propagated) over the output link, and forwards this information to the next router. Each router does the same. The last equipment (the server, in this case) is able to evaluate the typical delay packets will suffer. Based on that information, the receiver of the flow (the server) requests routers to reserve some bandwidth (R) for this flow. The RSVP RESV message is used for that purpose. After resources are reserved at the IP network elements, packets may start to be sent, having the bandwidth available and therefore a maximum guaranteed delay.

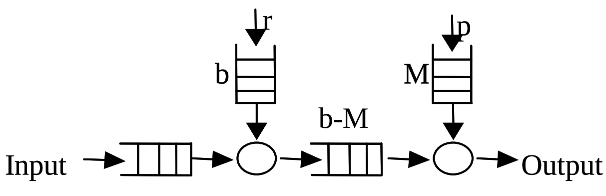


Figure 2-2 Traffic regulator

In order to be compliant with the information announced in the PATH message of RSVP, and get the maximum delay guarantees, the flow must be regulated by the mechanism shown in Figure 2-2. This mechanism consists of two buckets. Each bucket can contain up to a maximum number of tokens (b tokens for the first and M for the second). Each bucket is filled at a constant rate (r token/s for the former, and p token/s for the latter). Each token allows the transmission of one octet of data. When a packet arrives and reaches the head of the arrival queue, it can pass to the second queue only if, at least, the number of tokens available equals the length of the packet, including its header. If not, the packet has to wait for tokens. When a packet is transmitted towards the second queue, the tokens are removed from the first bucket. The second bucket has a similar behaviour. In this model, the first token bucket regulates the average transmission rate, and the second token bucket controls the maximum transmission rate.

From a router point of view, the support of IntServ consists in, after having accepted the flow, scheduling the packet in such a way that the bandwidth promised to the flow (bit/s) is available. Fair queuing techniques are used to this end, and they can provide to well behaved (compliant) flows the promised share of link level bandwidth.

The IntServ/RSVP approach is known to be not scalable and, for that reason, not usable at the core routers since they may have to support millions of simultaneous flows. In the access network the problem is not so serious in what concerns the number of flows. Wireless networks, however, bring other problems, not known in wired networks - the link level bandwidth and BER vary in time. In UMTS, for instance, the IP network elements (terminal and GGSN) may have to renegotiate, during a flow, the reservations made initially.

2.4.2 DiffServ

The DiffServ approach is more conservative and more easily implementable in IP network elements than IntServ. It can be also implemented independently in some portions of the IP network. In this approach, the old ToS field of the IP datagrams header is reused and renamed as DSCP (Differentiated Services Code Points). Packets at the entrance of a domain have this field marked according to some policy. The scheduler associated to a router output port or to the network interface of the terminal uses this field to prioritise the packets so that different classes may get different priorities and, for instance, a different percentage of the output port link bandwidth.

In any case, a common function set is required at any IP QoS network element. Arriving packets (from TCP in the terminal or from an input port in a QoS router) have to be classified and placed in a queue associated to the output port. The scheduler of each output port decides, from the set of packets available, the one that will be sent first, so that the quality negotiated by flow or by contract can be satisfied.

2.4.3 The IntServ over DiffServ approach

IntServ is implemented by "network elements". While it is common for network elements to be individual nodes such as routers or links, more complex entities, such as ATM "clouds" or 802.3 networks may also function as network elements. As discussed below, a DiffServ network (or "cloud") may be viewed as a network element within the larger IntServ network [15].

In the general framework shown in Figure 2-3, the Integrated Services architecture is used to deliver end-to-end QoS to applications. The network, however, includes some combination of IntServ capable nodes and DiffServ regions.

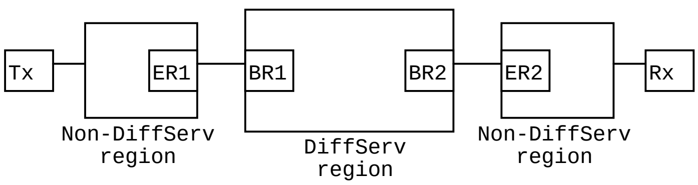


Figure 2-3 Sample Network Configuration

The reference network includes a DiffServ region in the middle of a larger network supporting IntServ end-to-end. We consider a single QoS sender (Tx) communicating across this network with a single QoS receiver (Rx). The edge routers (ER1, ER2) adjacent to the DiffServ region interface with the border routers (BR1, BR2) within the DiffServ region.

Both sending and receiving hosts use RSVP to communicate the quantitative QoS requirements of QoS-aware applications running on the host. Traffic control in the host may mark the DSCP in transmitted packets, and shape transmitted traffic according to the requirements of the IntServ service in use. Alternatively, the first IntServ-capable router downstream from the host may provide these traffic control functions. We assume that RSVP signalling messages travel end-to-end between hosts Tx and Rx to support RSVP/IntServ reservations outside the DiffServ network region. These end-to-end RSVP messages are carried across the DiffServ region.

Edge routers, ER1 and ER2 (e.g., GGSN in Figure 2-1), reside adjacent to the DiffServ network regions. Edge routers act as admission control agents to the DiffServ network. They process signalling messages from both Tx and Rx, and apply admission control based on resource availability within the DiffServ network region and on customer-defined policy. BR1 and BR2 are pure border routers, residing in the DiffServ network region. As such, their sole responsibility is to police the submitted traffic based on the service level specified in the DSCP and the agreement negotiated with the customer (Non-DiffServ region, aggregate traffic control).

The DiffServ network region supports aggregate traffic control and provides two or more levels of service based on the DSCP in packet headers. The network outside the DiffServ region consists of IntServ capable hosts and other network elements. IntServ service requests specify an IntServ service type. At each hop in an IntServ network, the IntServ service requests are interpreted in a form meaningful to the specific link layer medium. For example at an UMTS network, the IntServ service requests must be mapped into appropriate UMTS classes.

Requests for IntServ services are also mapped onto the underlying capabilities of the DiffServ network region. Aspects of the mapping include: 1) selecting an appropriate PHB (Per Hop Behaviour) for the requested service; 2) performing appropriate policing (including, perhaps, shaping or remarking) at the edges of the DiffServ region; 3) exporting IntServ parameters from the DiffServ region; 4) performing admission control on the IntServ requests that takes into account the resource availability in the DiffServ region. When the PHB has been selected for a particular IntServ flow, it is necessary to communicate the choice of DSCP for the flow to other network elements. RSVP may be used for that purpose.

Boundary routers residing at the edge of the DiffServ region will typically police traffic submitted from the outside the DiffServ region in order to protect resources within the DiffServ region. This policing will be applied on an aggregate basis, with no regard for the individual micro flows making up each aggregate. As a result, it is possible for a misbehaving micro flow to claim more than its fair share of resources within the aggregate, thereby degrading the service provided to other micro flows. This problem is addressed by providing per micro flow policing at the edge routers - this is generally the most appropriate location for micro flow policing, since it pushes per-flow work to the edges of the network, where it scales better. In addition, since IntServ-capable routers outside the DiffServ region are responsible for providing micro flow service to their customers and the DiffServ region is responsible for providing aggregate service to its customers, this distribution of functionality mirrors the distribution of responsibility.

2.5 UMTS QoS

3GPP proposes in [5] a framework for reasoning about QoS in UMTS networks. There, the end-to-end Service is considered between terminal equipments, and may have a certain Quality of Service (QoS).

To use a QoS network, a Bearer Service is set up from the source to the destination of the service. A bearer service includes aspects related to the provision of the contracted QoS, such as the control signalling, data transport and QoS management. In the UMTS bearer service layered architecture shown in Figure 2-4, each bearer service on a specific layer offers its services using the services provided by the layers below.

The *End-to-End Service* on the application level uses the bearer services of the underlying network(s). The *End-to-End Service* is conveyed over several networks. The *End-to-End Service* used by the TE is provided using a *TE/MT Local Bearer Service*, a *UMTS Bearer Service*, and an *External Bearer Service*. The UMTS Bearer Service provides the UMTS QoS. The UMTS Bearer Service consists of two parts, the Radio Access Bearer Service and the Core Network Bearer Service.

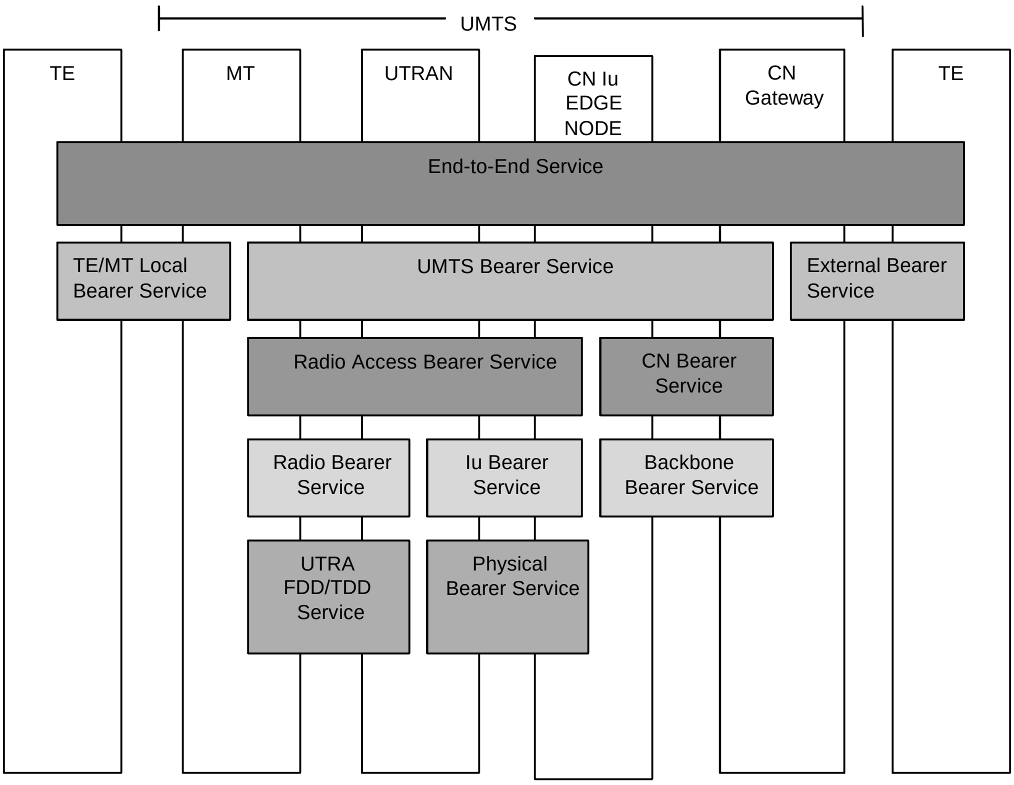


Figure 2-4 UMTS QoS Architecture

2.5.1 UMTS Bearer Service Attributes

Unidirectional and bi-directional bearer services are supported. For bi-directional bearer services, the attributes Maximum bit rate, Guaranteed bit rate, and Transfer delay should be set separately for uplink/downlink in order to support asymmetric bearers. The UMTS bearer service has the following attributes:

Traffic class ('conversational', 'streaming', 'interactive', 'background'). The type of application for which the UMTS bearer service is optimised.

Maximum bit rate (kbps). The maximum number of bits delivered by UMTS and to UMTS at a SAP within a period of time, divided by the duration of the period. The traffic is conforming to Maximum bit rate as long as it follows a token bucket algorithm where the token rate equals Maximum bit rate and the bucket size equals Maximum SDU size.

Guaranteed bit rate (kbps). The guaranteed number of bits delivered by UMTS at a SAP within a period of time (provided that there is data to deliver), divided by the duration of the period. The traffic is conforming to the Guaranteed bit rate as long as it follows a token bucket algorithm where the token rate equals Guaranteed bit rate and the bucket size equals Maximum SDU size. UMTS bearer service attributes, e.g. delay and reliability attributes, are guaranteed for traffic up to the Guaranteed bit rate. For the traffic exceeding the Guaranteed bit rate, the UMTS bearer service attributes are not guaranteed.

Delivery order (y/n) indicates whether the UMTS bearer shall provide in-sequence SDU delivery or not.

Maximum SDU size (octets). The maximum allowed SDU size.

SDU format information (bits). The list of possible exact sizes of SDUs.

SDU error ratio. The fraction of SDUs lost or detected as erroneous. SDU error ratio is defined only for conforming traffic.

Residual bit error ratio. The undetected bit error ratio in the delivered SDUs. If no error detection is requested, Residual bit error ratio indicates the bit error ratio in the delivered SDUs.

Delivery of erroneous SDUs (y/n/-). It indicates whether SDUs detected as erroneous should be delivered or discarded.

Transfer delay (ms). The maximum delay for the 95th percentile of the delay distribution for all delivered SDUs during the lifetime of a bearer service, where delay for an SDU is defined as the time from a request to transfer an SDU at one SAP to its delivery at the other SAP.

Traffic handling priority. It specifies the relative importance for handling of all SDUs belonging to the UMTS bearer compared to the SDUs of other bearers.

Allocation/Retention Priority. It specifies the relative importance compared to other UMTS bearers for allocation and retention of the UMTS bearer. The Allocation/Retention Priority attribute is a subscription attribute that is not negotiated from the mobile terminal.

Source statistics descriptor ('speech'/'unknown'). It specifies characteristics of the source of submitted SDUs.

For the UMTS Bearer service a list of finite attribute values or the allowed value range is defined for each attribute. The list/value range defines the values that can be used for an attribute considering every possible service condition for release 1999.

Table 2-1 Value ranges for UMTS Bearer Service Attributes

Traffic class	Conversational	Streaming	Interactive	Background
Maximum bit rate (kbps)	<2048	<2048	<2048-overhead	<2048-overhead
Delivery order	Yes/No	Yes/No	Yes/No	Yes/No
Maximum SDU size (octets)	<=1500 or 1502	<=1500 or 1502	<=1500 or 1502	<=1500 or 1502
SDU format information				
Delivery of erroneous SDUs	Yes/No/-	Yes/No/-	Yes/No/-	Yes/No/-
Residual BER	$5*10^{-2}$, 10^{-2}, $5*10^{-3}$, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}	$5*10^{-2}$, 10^{-2}, $5*10^{-3}$, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}	$4*10^{-3}$, 10^{-5}, $6*10^{-8}$	$4*10^{-3}$, 10^{-5}, $6*10^{-8}$

SDU error ratio	10^{-2}, $7*10^{-3}$, 10^{-3}, 10^{-4}, 10^{-5}	10^{-1}, 10^{-2}, $7*10^{-3}$, 10^{-3}, 10^{-4}, 10^{-5}	10^{-3}, 10^{-4}, 10^{-6}	10^{-3}, 10^{-4}, 10^{-6}
Transfer delay (ms)	100 – maximum value	250 – maximum value		
Guaranteed bit rate (kbps)	<2048	<2048		
Traffic handling priority			1,2,3	
Allocation/ Retention priority	1,2,3	1,2,3	1,2,3	1,2,3
Source statistic descriptor	Speech/unknown	Speech/unknown		

2.5.2 Non Access Stratum

The Non Access Stratum offers the UMTS Bearer Service to the upper layer, that is, to the IP Bearer Service. Each UMTS Bearer is described by a PDP Context. Figure 2-5, taken from [4], shows the Non Access Stratum protocol stack supporting the PS mode of operation at the terminal equipment side. The GGSN side is similar.

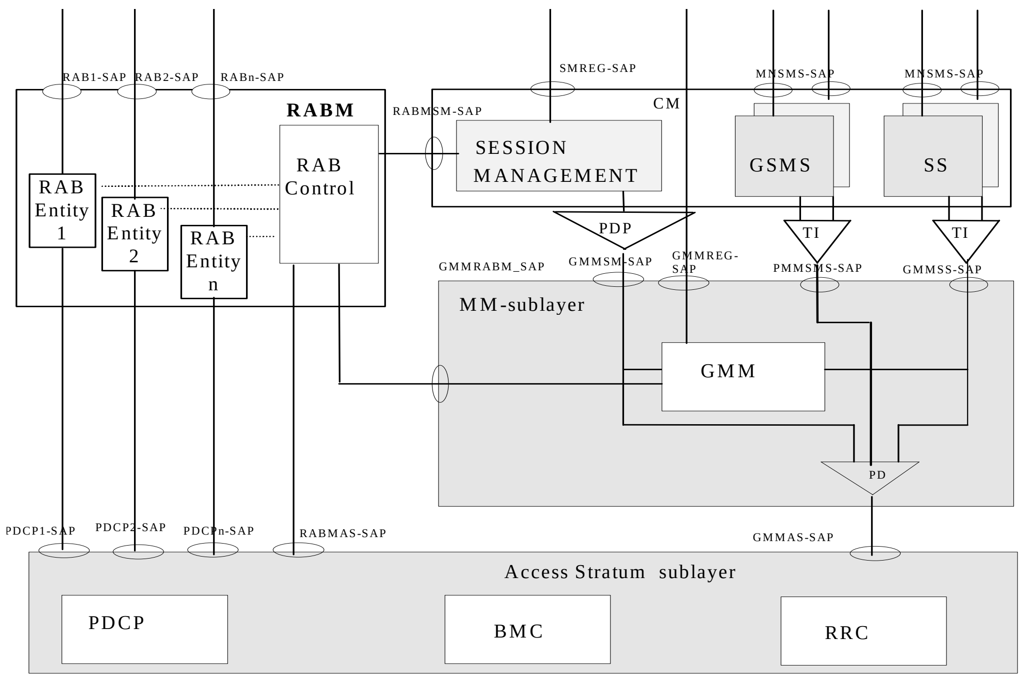


Figure 2-5 Protocol architecture of NAS supporting PS mode, Terminal Equipment side

The Session Management signalling interface (SMREG-SAP) available at the Terminal Equipment for activating, deactivating and modifying a PDP context is composed by the set of primitives presented in Table 2-2.

Table 2-2 Primitives and Parameters at SMREG-SAP - MS side

PRIMITIVE	PARAMETER (message, info elements of message, other parameters)
SMREG-PDP-ACTIVATE-REQ	PDP address, QoS, NSAPI, APN, Protocol configuration options
SMREG-PDP-ACTIVATE-CNF	PDP address, QoS, NSAPI, Protocol configuration options
SMREG-PDP-ACTIVATE-REJ	Cause, NSAPI, Protocol configuration options
SMREG-PDP-ACTIVATE-IND	PDP address, APN
SMREG-PDP-ACTIVATE-REJ-RSP	Cause, PDP address, APN
SMREG-PDP-DEACTIVATE-REQ	NSAPI(s) tear down indicator, cause
SMREG-PDP-DEACTIVATE-CNF	NSAPI(s)
SMREG-PDP-DEACTIVATE-IND	NSAPI(s) (s), tear down indicator, cause
SMREG-PDP-MODIFY-IND	QoS, NSAPI
SMREG-PDP-MODIFY-REQ	QoS, NSAPI, TFT
SMREG-PDP-MODIFY-CNF	QoS, NSAPI
SMREG-PDP-MODIFY-REJ	Cause, NSAPI
SMREG-PDP-ACTIVATE-SEC-REQ	QoS, NSAPI, TFT, Primary NSAPI
SMREG-PDP-ACTIVATE-SEC-CNF	QoS, NSAPI
SMREG-PDP-ACTIVATE-SEC-REJ	Cause, NSAPI

The Session Management signalling interface available at the GGSN for activating, deactivating and modifying a PDP context is composed by the set of primitives presented in Table 2-3.

Table 2-3 Primitives and Parameters at SMREG-SAP - network side

PRIMITIVE	PARAMETER (message, info elements of message, other parameters)
SMREG-PDP-ACTIVATE-REQ	PDP address, APN
SMREG-PDP-ACTIVATE-REJ	Cause, PDP address, APN
SMREG-PDP-DEACTIVATE-REQ	NSAPI(s), teardown indicator, cause
SMREG-PDP-DEACTIVATE-CNF	NSAPI(s)
SMREG-PDP-MODIFY-REQ	QoS, NSAPI
SMREG PDP-MODIFY-CNF	NSAPI
SMREG PDP-MODIFY-REJ	NSAPI

2.5.3 Policing at the UMTS Bearer Service

There is a difference between shaping and policing. Shaping is a function implemented by the user of a service with the main objective of regulating the traffic offered to this service. If a packet arrives and no tokens are available it must wait for the tokens before being delivered. Policing is a function implemented in the service. The service must know if a packet arriving to the service is compliant or not. If not compliant, the service is not required to serve it.

As referred to above, traffic arriving at a PDP Context SAP is policed by means of two independent token buckets – one for the maximum bit rate and the other for the guaranteed bit rate. Depending on the traffic class, the guaranteed bit rate token bucket may be considered or not. A packet being policed by a given token bucket may be classified as compliant or non-compliant to this bucket. A packet is compliant if, when it arrives, there are enough tokens for the packet. A packet is said to be non-compliant on the other case.

Policing in conversational and streaming classes. Policing in these classes use the two token buckets – the Guaranteed bit rate and the Maximum bit rate. Three outcomes may result from the policing operation: 1) the packet is non-compliant with the maximum token bucket – in this case the packet is eliminated; 2) the packet is compliant with the maximum token bucket but does not comply with the guaranteed token bucket – in this case, the number of tokens corresponding to the packet length is removed from the maximum token bucket and the packet is scheduled for transmission with no guarantees; 3) the packet is compliant with both the maximum token bucket and the guaranteed token bucket – in this case, the number of tokens corresponding to the packet length is removed from the two buckets and the packet is scheduled for transmission with guarantees.

Policing for the interactive and background traffic classes. Since no other guarantee than bit integrity in the delivery of data is given to the interactive and background classes, the guaranteed bit rate as well as the transfer delay are not defined. In this case, traffic is only policed by one token bucket – the maximum bit rate token bucket. Two outcomes may result from the policing operation, in this case: 1) the packet is non-compliant with the maximum token bucket – in this case the packet is eliminated; 2) the packet is compliant with the maximum token bucket – in this case, the number of tokens corresponding to the packet length is removed from the maximum token bucket and the packet is scheduled for transmission.

3 THE ARROWS END-TO-END QOS FRAMEWORK

IntServ was the model selected to provide end-to-end QoS. Its companion RSVP was selected for signalling. However, RSVP has a serious problem when working over wireless layer 2 technologies, such as UMTS – it cannot handle the QoS renegotiations initiated by the network. In order to overcome this problem, a management module was created for the mobile terminal - the *QoS Manager* - that handles these renegotiations for the applications, without modifying RSVP. It can, in some sense, be understood as the new RSVP end-point for mobile terminals.

In order to adapt applications to the new *QoS Manager*, in the mobile terminal, a new API was developed. This API, intended to be as simple as possible, can be easily used even by existing and non-QoS aware applications. In this way, the ARROWS architecture can use a number of well-established applications.

The *QoS Manager*, the *RSVP*, and the new QoS API, together, were used to overcome a known UMTS SAP (SMREG-SAP) characteristic that, for IP traffic, constitutes a problem - the SMREG-SAP has not the same primitives at TE and GGSN. From the GGSN side, for instance, it is not possible to ask for the activation of a PDP Context with QoS requirements. It implies that RSVP cannot ask for resource reservation at GGSN, for a flow towards the mobile terminal.

3.1 Multimedia Test Applications Architecture

Multimedia Applications are used in ARROWS [1][2][3] to evaluate, from a QoS perspective, the suitability of the new RRM algorithms defined by the project. For that purpose, multimedia applications were required to correctly implement the services identified in the project, and negotiate with the network the quality required for these application flows. The Multimedia applications were selected based on the UMTS QoS classes available:

1. Video telephony, for the conversational class;
2. Audio-video Streaming, for the streaming class;
3. Web Browsing, for the interactive class;
4. E-mail, for the background class.

Another generic requirement was that applications should generate traffic transportable as data packets, that is, IP datagrams. This fact led the ARROWS network demonstrator towards an all-IP scenario and the applications towards the IETF multimedia architecture.

The QoS architecture implemented in the ARROWS testbed is shown in Figure **3-1**, and it reflects the framework presented in Figure 2-4. The six functional blocks represented are (1) Application, (2) Non-Access Stratum, (3) RSVP, (4) QoS Manager (5) IP BS Manager and (6) IP Traffic Control. The lines connecting them represent paths through which information is conveyed. The bold lines, such as those connecting Application and Traffic Control, represent data paths. The other lines are used for signalling. Functional blocks are aggregated in different ways in TE, GGSN and Server network elements

.

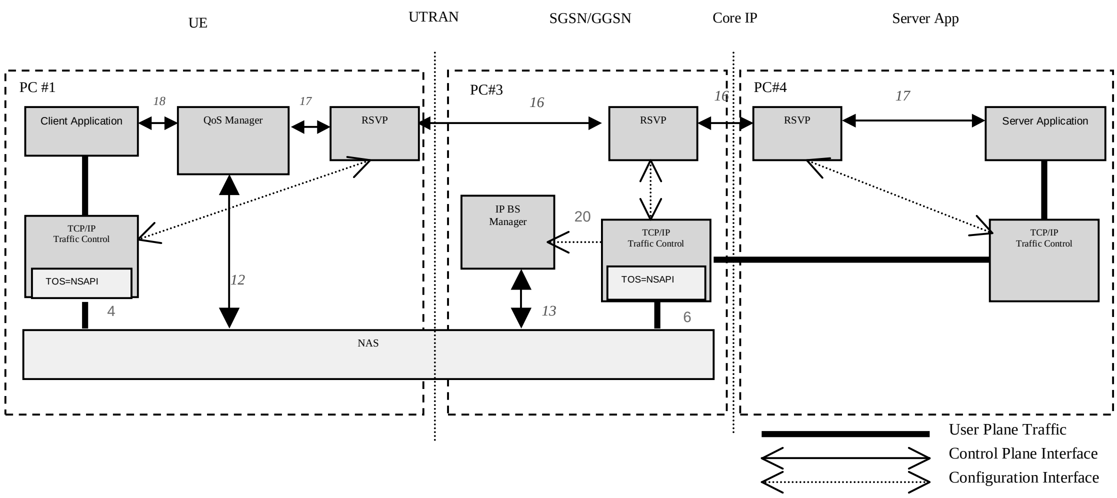


Figure 3-1 QoS architecture

Application. The application block represents a client/server process that communicates through UDP or TCP sockets and usually provides a well-known service. Services like WWW and mail use TCP, while the unicast video telephony as well as the audio-video streaming use UDP. These UDP services are real-time, use RTP and adopt the IETF multimedia service architecture.

Non-Access Stratum. This block, common to the edge UMTS network elements (terminal equipment and GGSN), represents the UMTS Non-Access Stratum, as defined in 3GPP standards. A block communicating with the Non-Access Stratum can activate, deactivate, modify or use (for packet transport) the PDP Contexts.

RSVP. This block, which represents the standard RSVP daemon, is responsible for handling IP reservations on an IP flow basis. It offers an API for the applications, that is, the sender and the receiver of Figure 3-1. In TE, this API is offered to the QoS Manager. The last interface allows RSVP to configure IP Traffic Control.

QoS Manager. This is the key block of the architecture. Besides allowing non-RSVP capable applications to use RSVP, the QoS Manager block is also in charge of (1) managing IP reservations, (2) activating/deactivating PDP contexts with QoS attributes, (3) mapping RSVP QoS parameters into UMTS QoS parameters and (4) deciding the multiplexing of IP flows into PDP contexts.

IP BS Manager. It exists in the GGSN and is responsible for requesting the activation of the Primary PDP Context (if not yet activated) upon the arrival of the first datagram to be delivered to the terminal equipment.

IP Traffic Control. After a PDP context is activated, the terminal may start transmitting datagrams. It is, however, necessary to direct the packets to the proper PDP Context, schedule the packets according to their priorities, and shape the traffic so that the flow is compliant with the QoS previously negotiated for that PDP Context.

3.2 The QoS Manager

The aspects that motivated the need for the QoS Manager functional block, in the terminal equipment, are described below.

SMREG-SAP asymmetry. SMREG-SAP is the NAS service access point for Session Management. Primitives for PDP context activation, deactivation and modification are exchanged at the SMREG-SAP [4][9]. This SAP exists at the UE and GGSN, but the primitives on each side are quite different. While UE primitives transport the QoS parameters, GGSN primitives do not. Therefore, unless other mechanism is used, a PDP Context with QoS can only be activated from the UE.

This means that IP cannot see the UMTS bearer service as a transparent link between the TE and the GGSN. As known, RSVP requests resources using the RESV message. This message is sent by the receiver of the IP flow and travels in the reverse direction of the flow. Since SMREG-SAP do not allow the activation of PDP Contexts with QoS at GGSN, the RSVP daemon at the GGSN cannot reserve the resource required for the downstream flow. In order to overcome this problem, first the IP layer reserves the resources between the GGSN and the remote host (server), and only after that the UMTS bearer (Secondary PDP Context) is established. The QoS Manager manages this process.

Dynamic availability of wireless resources. RSVP is unable to handle renegotiations initiated by layer 2, that is, RSVP is not dynamic. The QoS Manager also solves this problem. It receives the UMTS initiated request for modification of the Context and asks for modifications at the IP layer using the refresh property of RSVP. As known, the soft state concept forces RSVP to regularly refresh the reserve. Then, if acceptable for the applications, the new RSVP reserve message describes the new conditions of the wireless link.

QoS parameters mapping and flow aggregation. The QoS Manager is responsible for mapping IntServ QoS parameters (TSPEC, RSPEC) into and the PDP Context QoS attributes. By controlling this aspect, QoS manager also becomes the right component for multiplexing or aggregating IP flows into PDP Contexts.

Control at the mobile terminal. The location of the QoS Manager at the mobile terminal allows the user – the client who pays the wireless services – to control the services requested to both the UMTS access network and the IP external network.

RSVP aware applications. In ARROWS, it was decided that all the applications use the new QoS API. A trivial but not implemented extension could be to have the QoS Manager providing also the RAPI (RSVP API) [7] to standard RSVP applications running in the mobile terminal equipment.

3.3 Signalling example

Figure3-2 shows the messages exchanged during the establishment of a video streaming session, where two IP flows and two Secondary PDP Contexts are used.

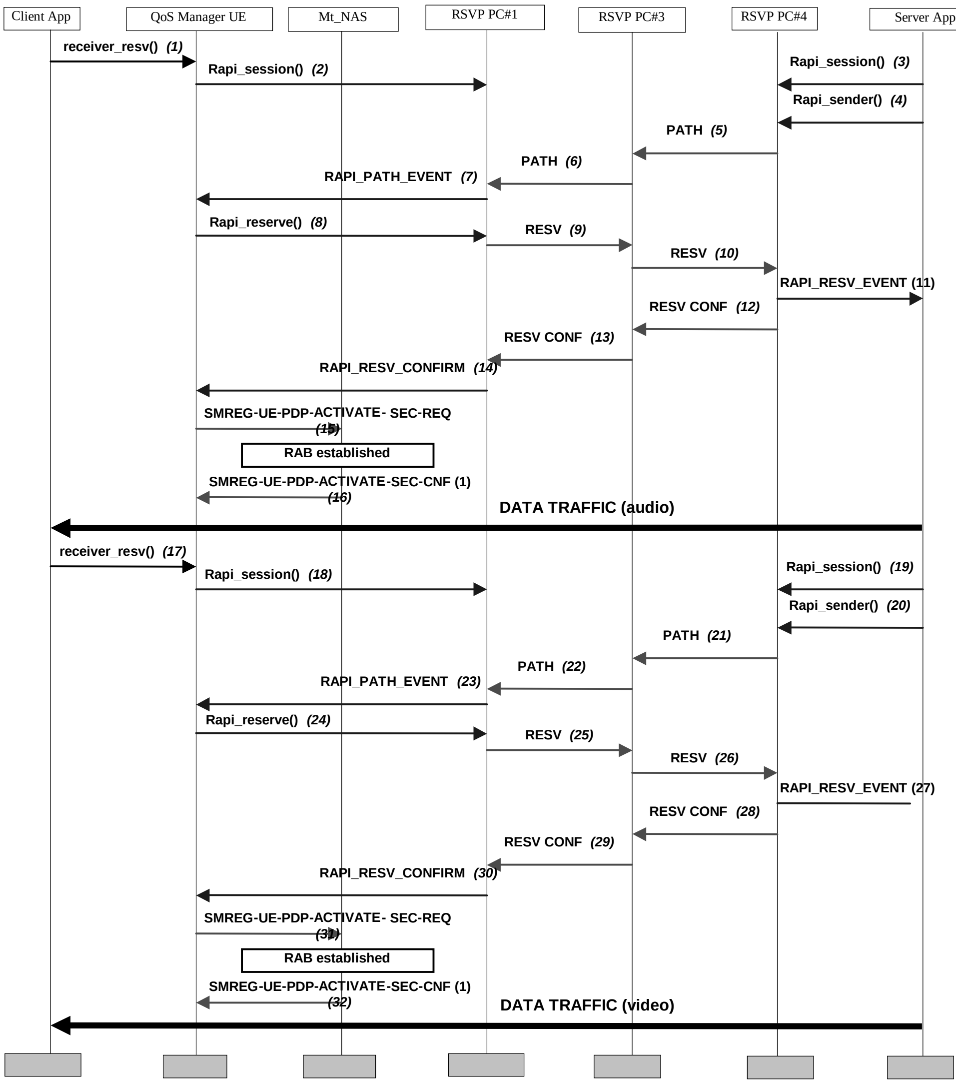


Figure3-2 RSVP and PDP Context signalling process

3.4 Simple TFT

3GPP defines a mechanism for filtering packets at the UMTS Bearer - the *Traffic Flow Template* (TFT). In ARROWS, a simpler approach was used – IP datagrams are filtered only based on their ToS field.

The datagrams are marked in the IP layer, at the entry side of the UMTS Bearer (terminal or GGSN). In order to let GGSN know about the ToS value of a given flow, a new Information Element was added to RSVP. This Information Element needs to be interpreted only by the RSVP daemons located at the Terminal Equipment and at the GGSN. For simplicity, in ARROWS, this new Information Element transports the NSAPI value associated with the UMTS bearer through which the IP datagram is transported.

Aggregation of multiple IP flows into the same PDP Context becomes also simpler – only the ToS field of the packets needs to be modified.

3.5 The QoS API

The QoS API was designed so that applications do not need to wait for responses from the network - the QoS Manager handles them. The application specifies its QoS requirements before the flow starts (before the TCP connection establishment or when the first UDP datagram is sent) and when the reservation is to be released (e.g., closing of the socket).

Table 3-1 presents the API function prototypes, while Table 3-2 gives a brief description of the service they provide.

Table 3-1 API prototypes

RETURN	FUNCTION	PARAMETERS
SESSION_ID (accepted) or -1 (rejected) or -2 (no service manager)	sender_resv()	SUBSERV, DEST_ADDR, DEST_PORT, SRC_ADDR, SRC_PORT, PROTOCOL, QOS
RECV_ID (accepted) or -1 (rejected) or -2 (no service manager)	receiver_resv()	SUBSERV, DEST_ADDR, DEST_PORT, PROTOCOL, QOS
SESSION_ID (accepted) or -1 (rejected) or -2 (no service manager)	client_resv()	SUBSERV, SERVER_ADDR, SERVER_PORT, CLIENT_ADDR, CLIENT_PORT, PROTOCOL, QOS
RECV_ID (accepted) or -1 (rejected) or -2 (no service manager)	server_resv()	SUBSERV, SERVER_ADDR, SERVER_PORT, PROTOCOL, QOS
void	release()	RECV_ID

The functions sender_resv() and receiver_resv() are intended to be used for simplex, connectionless flows, such as UDP. On the other hand, client_resv() and server_resv() are meant to be used for duplex, connection-oriented flows, such as TCP.

Table 3-2 Description of the functions

FUNCTION	DESCRIPTION
sender_resv()	This function is issued by the application (sender) to request resources reservation for a simplex UDP flow. Parameters include the sub-service, source and destination addresses and ports, and QoS parameters.
receiver_resv()	This function is issued by the application (receiver) to notify the *QoS Manager* that it is waiting for a flow. Parameters include the sub-service, destination address and port, and QoS parameters. Source address and port are not required in this case, since the QoS Manager will discover them when it receives a PATH message.
client_resv()	This function is issued by the application (client) to request reservation of resources for a duplex flow. Parameters include the sub-service, source and destination addresses and ports, and QoS parameters.

server_resv()	This function is issued by the application (server) to notify the *QoS Manager* that it is waiting for a connection. Parameters include the sub-service, destination address and port, and QoS parameters. Source address and port are not required in this case, since the QoS Manager will discover them when it receives a PATH message.
release()	This function sends an indication that the application wants to tear down all resources related to the specified flow.

Using the sub-service parameter, an application can pass to QoS Manager additional information about the flow.

Table 3-3 Sub-services relation with QoSManager

VALUE	DESCRIPTION
UNKNOWN	The QoS Manager should attempt to guess the subservice based on source or destination port
WEB	All TCP flows are to be carried into the same PDP Context.
E-MAIL	As with the WEB sub-service, all TCP flows will be carried into the same PDP Context
VT_VIDEO	In telephony, the video stream, which is transported over UDP, is carried into the same PDP Context as VT_AUDIO. There is one PDP Context for each direction (UE->Server and Server->UE)
VT_AUDIO	The audio stream, also transported over UDP, is carried into the same PDP Context as VT_VIDEO. There is one PDP Context for each direction (UE->Server and Server->UE)
VSTR_BASE	The audio base layer stream in the audio- video Streaming Service is delivered over UDP, and carried in one PDP Context from the Server to the UE.
VSTR_ENH	The video layer stream in the Video Streaming Service is delivered over UDP, and carried in one PDP Context from the Server to the UE. This PDP Context is not the same as to VSTR_BASE.

Table 3-3 presents the values for the sub-services used in ARROWS. In this case, the QoS Manager multiplexes the audio and video IP flows into the same Secondary PDP Context. However, any policy for multiplexing IP flows into PDP Contexts can be configured.

In ARROWS, a simple approach was selected for reserving bandwidth – for real-time services, only constant bit rate contexts are negotiated. Thus, the QoS parameter needs only to carry out information about the rate required by the application. Using this information, the QoS Manager sends the RSVP PATH message with p=r and b=m =MTU. More flexible, but also more complex to implement approaches, could be implemented – the sender application, for instance, could pass TSPEC parameters into the QoS parameter; based on that, the QoS Manager could calculate the reserved bandwidth R, when the application in the mobile is the flow receiver.

With this API, applications become simple to extend since they do not receive responses from the QoS Manager. On other hand, the application is also unaware of renegotiations. In ARROWS, only the specific audio-video streaming application was explicitly developed to react to renegotiation requests. The other applications use TCP and RTP+RTCP mechanisms to adapt their traffic to the available bandwidth. TCP provides the congestion control mechanism. RTCP issues message reports that are used by applications to adapt their rate.

3.6 QoS Mapping

In ARROWS, RSVP signalling as well as best effort traffic is carried out over the Primary PDP Context. QoS demanding flows are transported over secondary PDP contexts. Although in ARROWS all the four UMTS classes of service are used, in this section only the conversational and streaming classes will be addressed. These two classes have the same QoS attributes: Transfer Delay, Maximum Bit Rate, Guaranteed Bit Rate and Maximum SDU Size.

Transfer Delay. The RSVP PATH message must carry information about the transfer delay of a message, which is sent in parameter Di. In the UMTS terminal equipment, Di is the link layer transfer delay for the UMTS bearer, between the TE and the GGSN. In ARROWS, the approach selected was to assume pre-defined values for Transfer Delays that depend on the traffic class.

Guaranteed Bit Rate, Maximum Bit Rate, Maximum SDU Size. For the Guaranteed IntServ service there are two relevant parameters: 1) TSPEC, based on r, b, p, M, m, that specifies the flow; 2) RSPEC, based on the R parameter, indicating the bandwidth to reserve. Considering that (1) the Guaranteed IntServ service class expects link layers to have no or low packet losses, (2) UMTS bearers do not guarantee the transport of packets not compliant with the Guaranteed bit rate, (3) the UMTS bearer policing mechanism is based on a token bucket model, supporting a burst of M (Maximum SDU Size) tokens, the solution adopted was the following:

- reshaping the IP traffic at each entry SAP, with R=p=r, and b=M;
- setting Guaranteed Bit Rate = Maximum Bit Rate = p.

This simple solution has two drawbacks: 1) buffers are required at the traffic flow generator to accommodate short bursts; 2) in this situation, the new maximum end-to-end delay is that calculated by IntServ plus the time packets have to wait in these queues. In ARROWS, the buffer sizes were calculated experimentally in the testbed.

3.7 IP Traffic Control

Figure 3-3 shows the traffic control configuration used in ARROWS, which implements the IETF IntServ managed by RSVP (RSVP daemon).

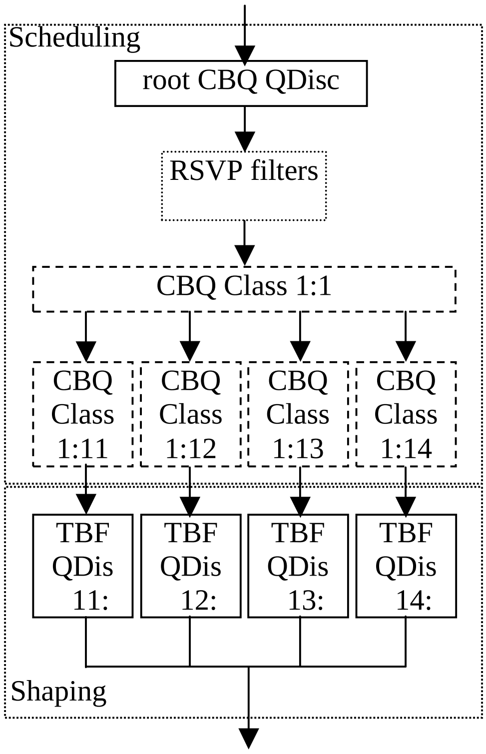


Figure 3-3 IP Traffic Control

Scheduling and shaping of the flows are implemented with CBQ (Class Based Queuing) [10] and TBF (Token Bucket Flow) [14] disciplines, respectively. It is also shown the location of the filters. For each PDP Context, one leaf class on the CBQ queuing discipline (e.g. 1:11) is created. Moreover, each class has one TBF queuing discipline associated instead of the generic one installed by default.

For each RSVP reservation, the RSVP daemon creates an instance of a CBQ class and respective TBF Queuing discipline (IETF Token bucket traffic regulator). In ARROWS, the latter was modified in order to (1) mark all the IP datagrams that pass through it with the appropriate ToS value, and (2) at TE and GGSN, have a bigger buffer for reshaping.

3.8 Primary and Secondary PDP Contexts

The ARROWS approach establishes initially a Primary PDP context for the best effort traffic class, and Secondary PDP Contexts for IP flows having explicit QoS requirements. Signalling, control and the best effort traffic, not having specific QoS requirements, are transported through the Primary PDP context. The Primary PDP Context was selected in ARROWS to be a background traffic class. In alternative, an interactive class could be used.

QoS mapping happens solely between IntServ/RSVP named flows and Secondary PDP Contexts. The QoS Manager can map one IP flow into one Secondary PDP Context or multiple IP flow into one Secondary PDP Context. For the second case, pre-definitions are required for the QoS Manager. In ARROWS, for instance, the terminal equipment multiplexes four conversational IP flows into one conversational Secondary PDP Context.

3.9 UMTS bearers

Table 3-4 shows the UMTS Bearer attributes used in ARROWS, along with the applications selected to load it.

Table 3-4 PDP Context definition in ARROWS Project

	Audio telephony		*Audio-video Streaming*		*Web Browsing*		*Email*		*Signaling (Primary PDP Context)*	
Traffic class	**Conversational class UL/DL**		**Streaming class (only DL)**		**Interactive class**		**Background class**		**Background class**	
			Base L.	**Enh. L.**	**UL**	**DL**	**UL**	**DL**	**UL**	**DL**
Maximum bitrate (kb/s)	64		32	16, 32, 48, 64, 80, 96	64	256	32	32	32	32
Delivery order	Yes		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes y
Maximum SDU size (octets)	570		570	570	570	570	570	570	570	570
SDU format information	-		-	-	N/A	N/A	N/A	N/A	N/A	N/A
Delivery of erroneous SDUs	Yes		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Residual BER	10^{-6}		10^{-6}	10^{-6}	$6*10^{-8}$	$6*10^{-8}$	$6*10^{-8}$	$6*10^{-8}$	$6*10^{-8}$	$6*10^{-8}$
SDU error ratio	10^{-5}		10^{-5}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
Transfer delay (ms)	100		1000	1000	N/A	N/A	N/A	N/A	N/A	N/A
					60, 120, 200					
Guaranteed bit rate	64		32	0	N/A	N/A	N/A	N/A	N/A	N/A
					(32, 64) (16, 64) (16, 32)					
Traffic handling priority	-		-	-	-	-	-	-	-	-
Allocation/Retention priority	-		-	-	-	-	-	-	-	-
	Audio	Video								
b	M		M	M	M	M	M	M	M	M
r	16	48	32	16, 32, 48, 64, 80, 96	64	256	32	32	32	32
p	16	48	32	16, 32, 48, 64, 80, 96	64	256	32	32	32	32
M	570		570	570	570	570	570	570	570	570
R	16	48	32	16, 32, 48, 64, 80, 96	64	256	32	32	32	32

4 RESULTS OBTAINED IN THE LAB

The developments carried in ARROWS concerning the Multimedia Test Applications are mainly related to the negotiation of quality of service contracts and to the shaping of the flows using these services. Therefore, it is of primordial importance that the overall system can be validated with respect to these aspects, which consist in observing whether the offered flows are compliant and if they receive the service contracted, usually described based on delay and reliability statistical estimators. In order to evaluate these aspects, a test tool was developed, which captures the data being transported as IP datagrams, analyses it, provides the relevant results and displays the graphics required to reason about the system/traffic behaviour.

The following aspects related to the transport of data packets are evaluated:

- **IP flow shaping**. Each application generates a set of IP flows, where an IP flow is a set of similar packets close in time. All the packets of an IP flow have the same IP source address, IP destination address, transport protocol, source port and destination port. In order to get a maximum guaranteed delay, each IP flow needs to be shaped and reshaped, if necessary, in order to comply with the Token Bucket (b, r, p, M) specification negotiated by RSVP.
- **UMTS Bearer Service traffic compliance**. Another aspect to verify is the relationship between layer 3 (IP) and layer 2 (UMTS Bearer Service) at the UMTS network elements. For each application, one or two (Video Streaming, only) Secondary PDP contexts are used. Depending on the application, a number of IP flows can be transported in a Secondary PDP Context (one for audio/video Streaming, two for Web and E-mail, four for video telephony). When activated, a Secondary PDP context is negotiated as a set of QoS parameters. This contract implies that the traffic arrives from IP to a Secondary PDP context according to certain characteristics. These characteristics are supposed to be policed by the Secondary PDP Context by the two independent Token Bucket previously mentioned. The compliance of the traffic offered to the UMTS bearers must also be evaluated.
- **Service received from Secondary PDP context**. For compliant traffic, an UMTS bearer is supposed to transport it adequately, that is, (1) to provide at least the guaranteed bit rate, (2) to not delay the compliant packets more than the transfer delay negotiated, and (3) to not lose or corrupt more packets than the negotiated. The service offered by the secondary PDP Context must then be evaluated, as well.

4.1 Test Architecture

The service test architecture is presented in Figure 4-1. There, we can observe the three network elements dealing with IP - TE, GGSN and the SERVER. IP is here assumed to be the end-to-end service mentioned in [5]; IP uses Secondary PDP Contexts in the UMTS network. IP flows are end-to-end packet flows and, in UMTS, they can be multiplexed in a PDP Context. The regulator R shapes each flow entering a PDP context.

4.1.1 Analyser tool

The tool developed analyses the IP datagrams exchanged through the four network interfaces of the testbed (one at TE, two at GGSN and one at the SERVER). As input, the analyser uses these packet files as well as signalling exchanged between relevant modules. After processing the signalling and the packet logs, the analyser provides information about every IP flow observed and Secondary PDP context used. In addition to the numeric results, the tool can also display packet arrival and packet delay plots for all IP flows.

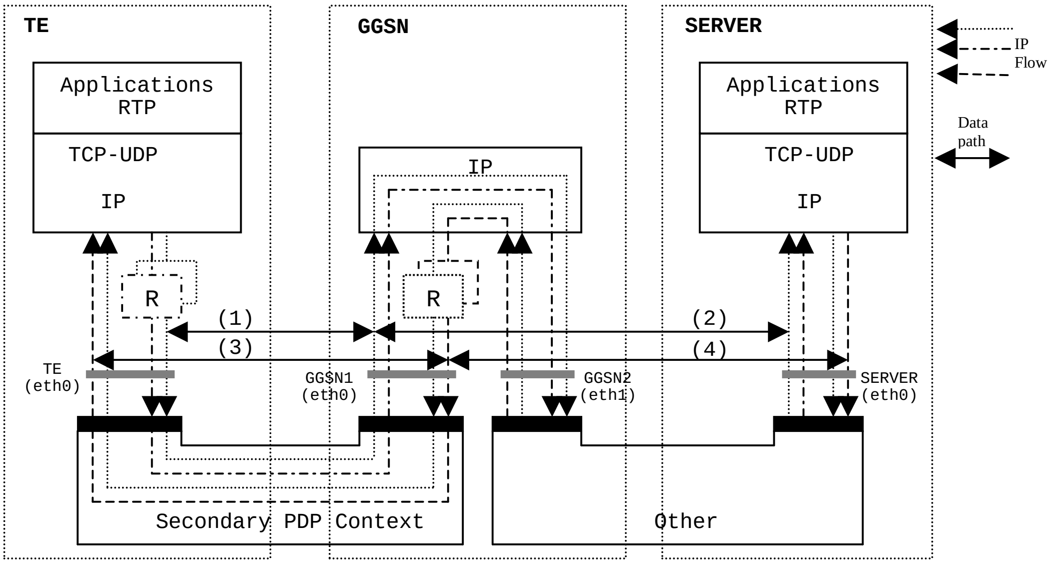


Figure 4-1 Service Test Architecture

4.1.2 Packet arrival plots

A packet arrival plot is basically a plot of accumulated packet sizes over time in a given capture point. An arriving packet is represented as a step in the plot, with vertical distance corresponding to the packet size. Although we call it packet *arrival*, sometimes it can really be packet *departure*. The analyser only sees packets appearing in a network interface. The packet may appear at such an interface either because it is being received or because it is about to be transmitted.

Packet arrival curves can show the shape of packet flows being transmitted or received. The slope of such a curve gives an indication of the data rate. More important, if one represents two packet arrival curves on two different network points in the same plot, the following can be concluded: 1) horizontal distances between the two curves are and indication of packet delays between those two points; 2) packet losses manifest as a down vertical shift on the curve corresponding to the most downstream capture point.

4.1.3 Delay plots

Delay plots are a simple graphical representation of the data structures constructed by the analyser to compute packet delays statistics. This information is collected by comparing packets on two different capture points and taking the difference of time stamps.

Delays are especially problematic due to the inherent imprecision of PC clocks. Not only do the PC's clocks need to be synchronized, they have to remain synchronized during the trial. Unfortunately, their drift is too large. Because of this problem, during trials network synchronization programs have to be used. The synchronization programs keep running in background, and help keep the clocks synchronized. Unfortunately, they also introduce "noise" in the clocks, due to frequent updates to the clock. Synchronization periods of one second and half a second were used. In these trials, errors of about 2-3 milliseconds are estimated.

4.1.4 Data paths and capture points

The ARROWS testbed is completely asymmetrical. In particular there are differences in results obtained in uplink flows and downlink flows. Figure 4-1 indicates "data paths", i.e. pairs of capture points used to compute delays. The following data paths are considered:

1. This is between interfaces TE and GGSN1, for upstream flows. It measures only the UTRAN (UMTS) delay. Although there is reshaping in upstream flows, the capture point ‘TE’ is beyond reshaping. For that reason, upstream reshaping is not measured;
2. This is between interfaces GGSN1 and SERVER, for upstream flows. It measures the routing delay of GGSN and core network delays;
3. This is between interfaces GGSN1 and TE, for downstream flows. It measures only UTRAN delay;
4. This is between interfaces SERVER and GGSN1, for downstream flows. When packets are captured in GGSN1, they have already been subject to reshaping.

The important conclusion is that the analyser can see the effect of reshaping delay only for downstream flows. Upstream flows also have reshaping, but it goes undetected by the analyser.

4.1.5 UTRAN emulation

Between the TE and GGSN PCs, there is in ARROWS a PC used to emulate the UTRAN. This emulation can be achieved naturally in the ARROWS testbed, using the ARROWS modules developed for that purpose. These modules were still under development at the time of this writing, so an alternative to UTRAN emulation has been used. The role of UTRAN emulator was fulfilled by a software tool called NIST Net [29].

NIST Net is a network emulation package that runs on Linux. It allows configuring packet filters associated with network performance scenarios. Filters can be source/destination IPs, together with ports and protocols, and/or ToS values. Selecting by ToS values is particularly useful in ARROWS. Network performance scenarios include bandwidth limits, delay distributions, and packet drops, among others.

For each service test, there is one trial without UTRAN and another with UTRAN emulation. For the trial with UTRAN emulation, NIST Net is configured to filter packets that would normally go into a PDP Context, and apply to them a bandwidth limit and a delay taken from the PDP Context QoS field. This allows us to roughly evaluate how a UMTS link would affect the end-to-end service offered to the applications.

4.2 Video telephony

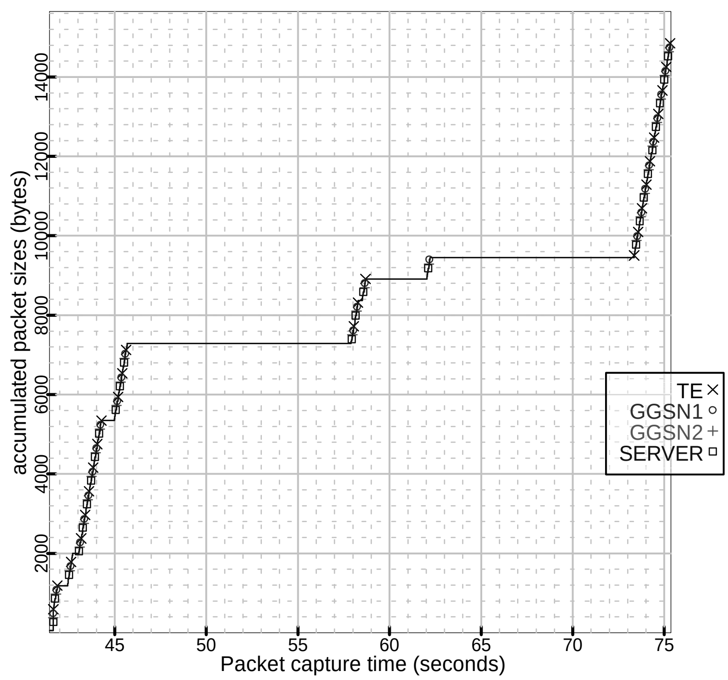
This service allows two users to communicate with audio and low quality video. One of the users is located in the server computer and the other in the terminal equipment. When the service is used, four unidirectional IP flows are generated (two for audio and two for video). In the UMTS network, these flows are multiplexed and transported in one bi-directional, conversational, PDP Context (UMTS Bearer Service).

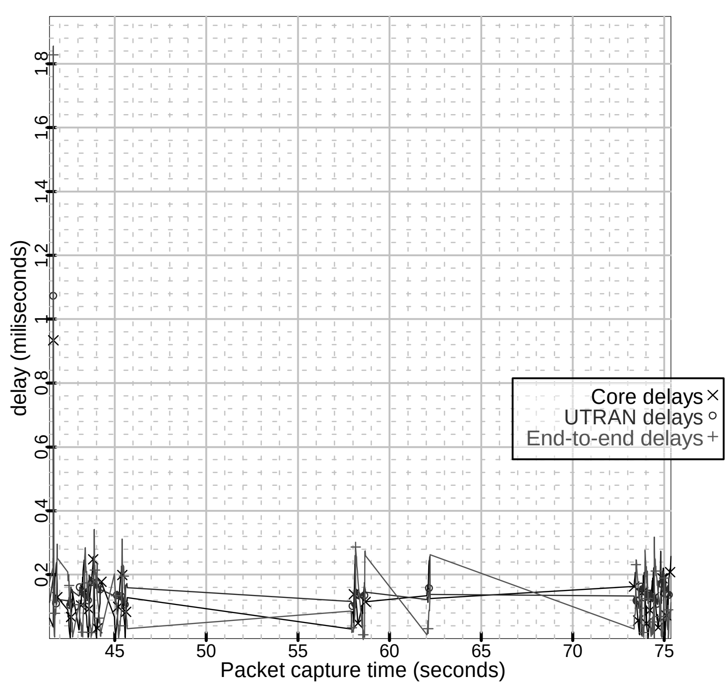
Video-telephony is based on VIC and RAT [26][27]. The video and audio streams are transported over RTP/UDP. The UTRAN / UMTS facilities are not used. The purpose of this demonstration is to obtain results that can be used to evaluate the effect of the UTRAN, in a subsequent test. It serves also for evaluating the video telephony implementation separately, without interference from the UTRAN emulator.

4.2.1 Without UTRAN

In this trial, VIC and RAT run with QoS reservation. However, this trial is done without UTRAN restrictions. Therefore, PC#2 becomes a simple IP router using only Ethernet layer 2 technology (100 Mbit/s). At IP QoS level, there is one RSVP reservation for each flow: audio-uplink, audio-downlink, video-uplink, and video-downlink.

4.2.1.1 Audio flow, downlink



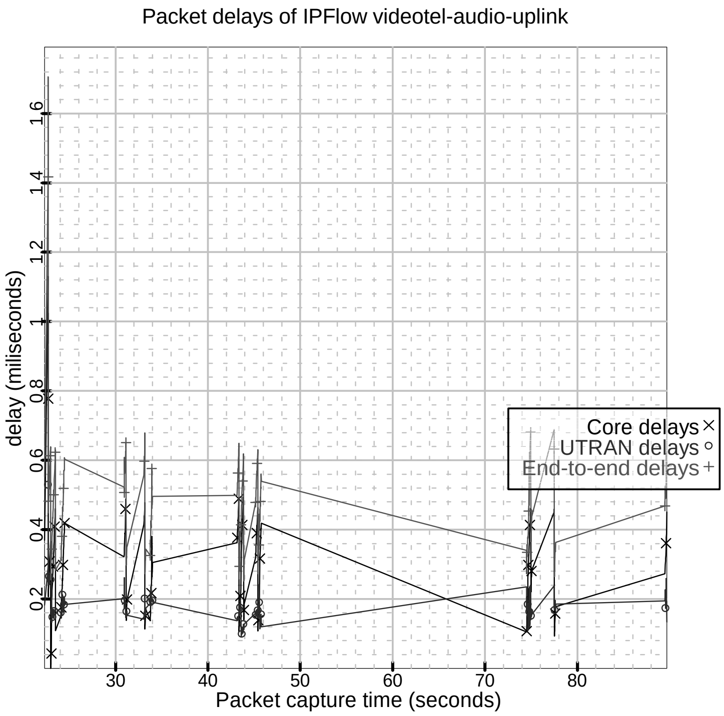


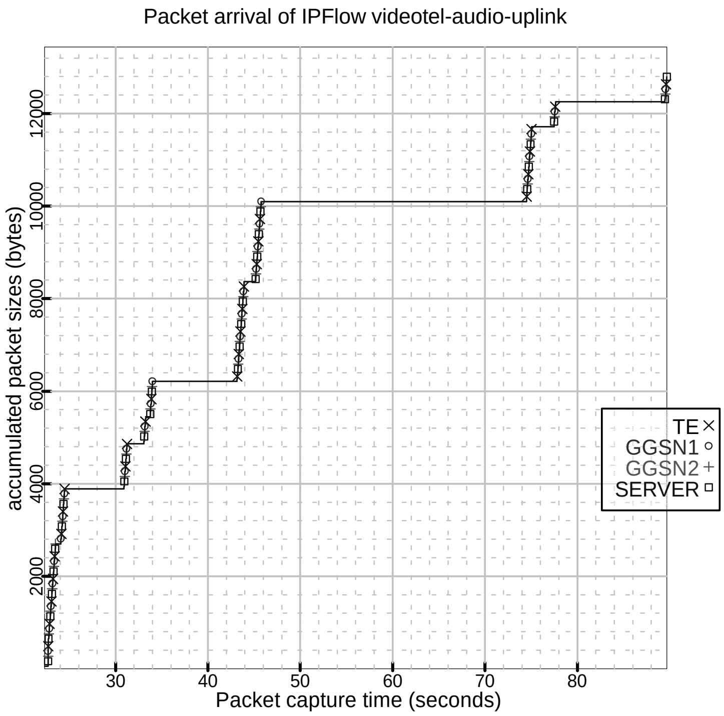
These plots refer to the downlink (Server to TE) audio stream of telephony. In the packet arrival plot, we can observe the typical service curve of an audio conversation. There are two fundamental types of periods. One is characterized by a high and steady slope, and corresponds to the period when the user is talking. The other one corresponds to the inactivity periods, when the user is not talking at all – only listening to its conversation peer. In this case, the audio codec automatically detects silence periods, and stops generating packets, which results in a flat, horizontal region of the curve.

Also, from the near superposition of the four service curves we can conclude beforehand that all delays are very low. This prediction can be confirmed by looking at the delays plot.

Although that plot looks a bit chaotic, that is understandable given the very low values represented. Any value below 2-3 ms is very much error prone because of the poor clock synchronization between the testbed PCs.

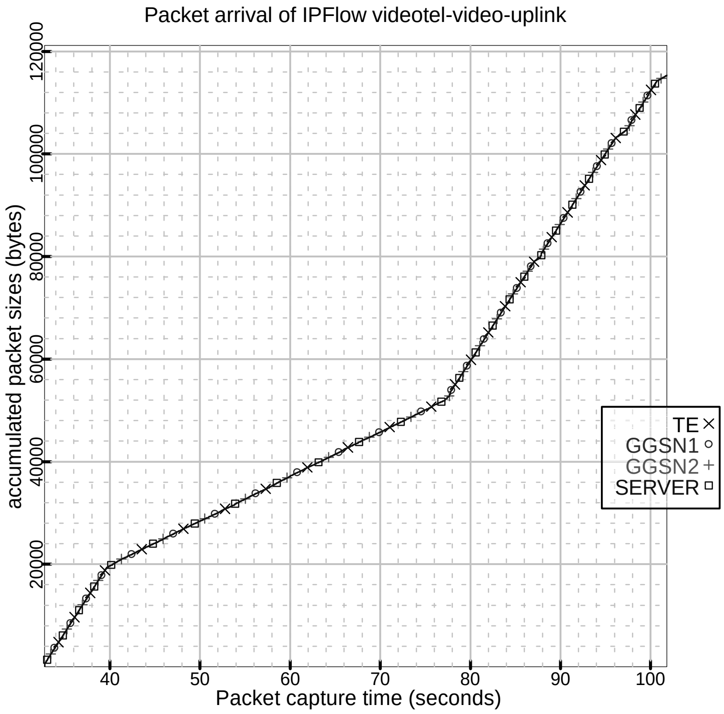
4.2.1.2 Audio flow, uplink

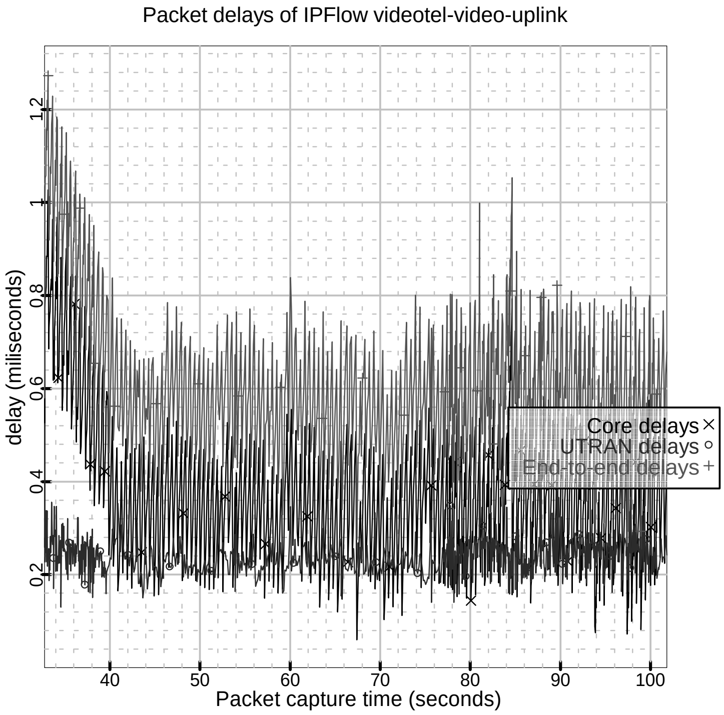




The two plots above refer to the uplink (TE to Server) audio stream of video telephony. They look pretty much identical to the downlink case, and the same conclusions apply here as well.

4.2.1.3 Video flow, uplink

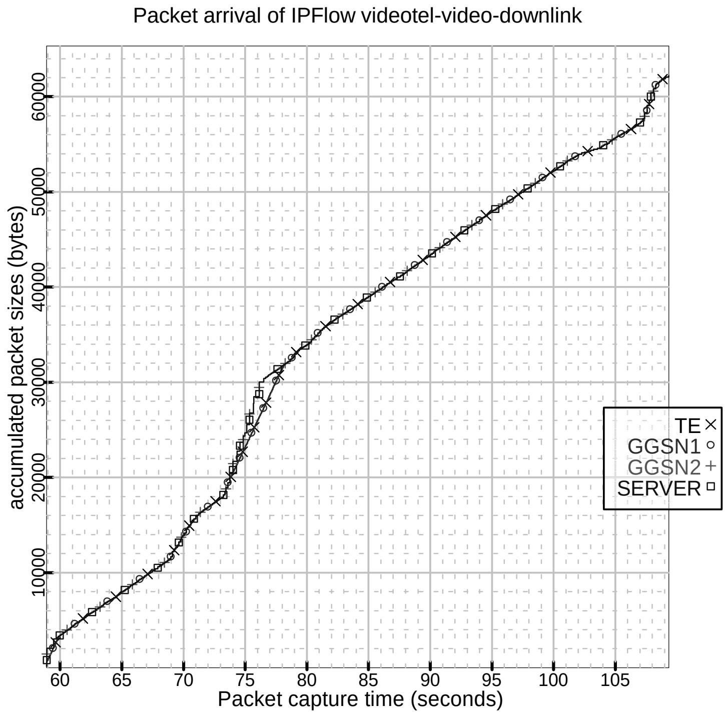


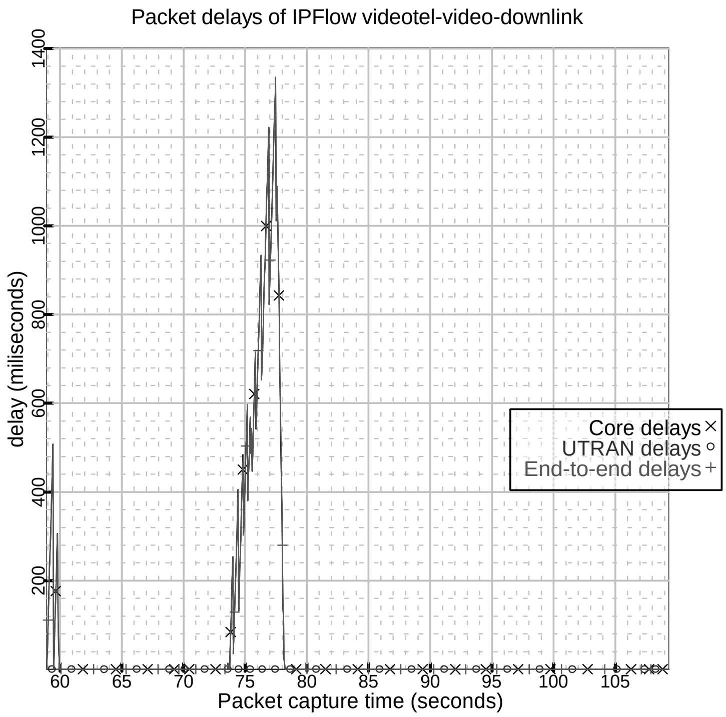


The two plots above refer to the uplink (TE to Server) video stream of video telephony. In the service curve plot, we can clearly distinguish two different slopes. This means that VIC generates, in this instance, roughly two distinct bit rates. This can be explained intuitively with some understanding of how the H.263 codec, used in these tests, works. The codec divides each video frame in blocks. The codec transmits, for each frame, the blocks that have changed from the previous frame. When the webcam captures a rapidly changing video, there is a lot of changing image blocks, so there is a lot of information to transmit, hence the higher slope of the service curve in some areas. However, when webcam is capturing a mostly still video, the number of blocks to update is very low, and so is the generated data stream.

Regarding the delays, they are very low, so the plot is very "noisy". It is important to remember that the uplink test cases cannot represent the RSVP reshaping effect.

4.2.1.4 Video flow, downlink



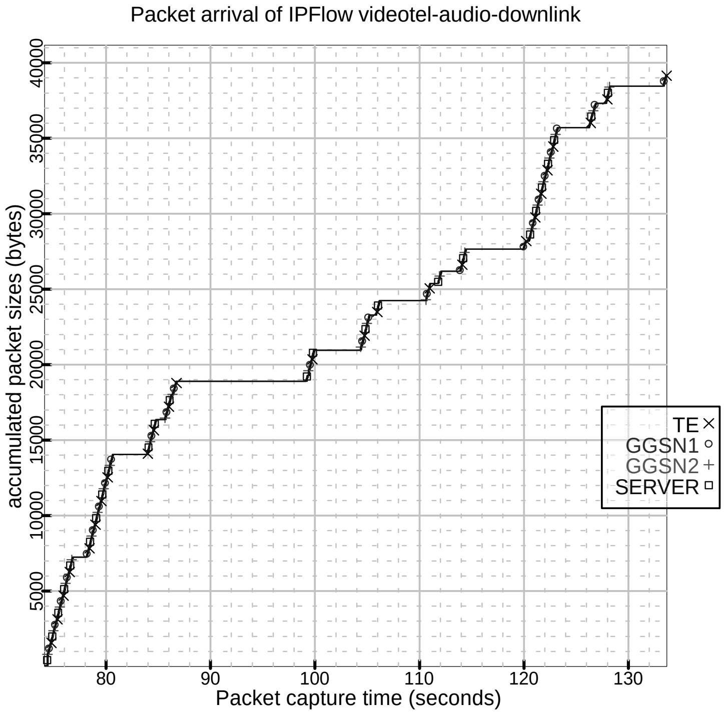


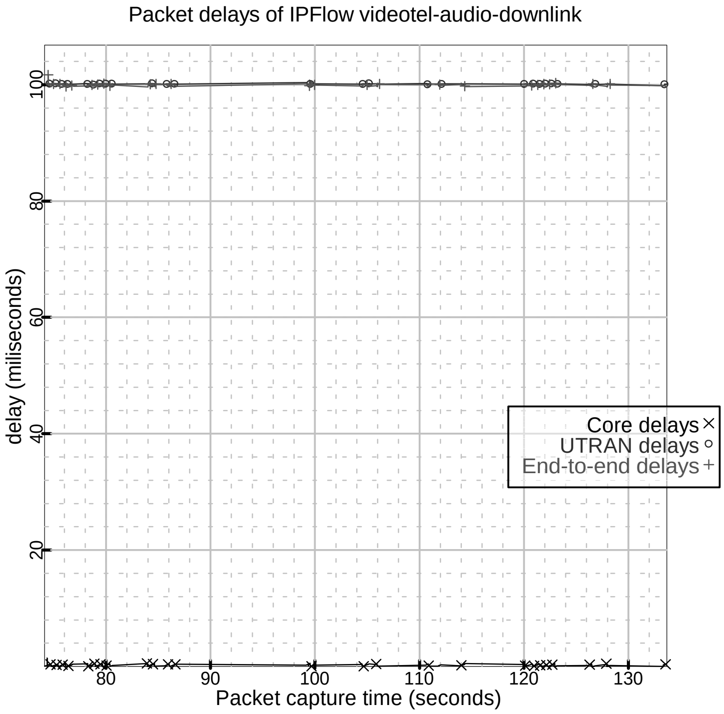
The two final plots refer to the downlink (Server to TE) video stream of video telephony. The most important aspect in this case is the effect of RSVP reshaping. This is clearly visible in the delay plot, as a peak at about second 76. It is also perceptible in the service curves; the curves grow apart, with SERVER and GGSN2 taking one path, and GGSN1 and TE taking a distinct path, and rejoining later on, as delay decreases. The peak value of ~1.2 s is the result of the application generating more traffic than the network can handle, for too long.

4.2.2 With UTRAN emulation

The purpose of this demonstration is to show the impact of the selected PDP Context on video telephony. VIC and RAT applications run with IP QoS reservations, over the conversational PDP Context. They register with the QoS Manager. NistNet is used to emulate the PDP context. NistNet is configured to filter packets with ToS 0x04, and apply to them a delay of 100 ms and bandwidth limit of 8 kbyte/s (= 64 kbit/s). This is in accordance with the negotiated PDP context.

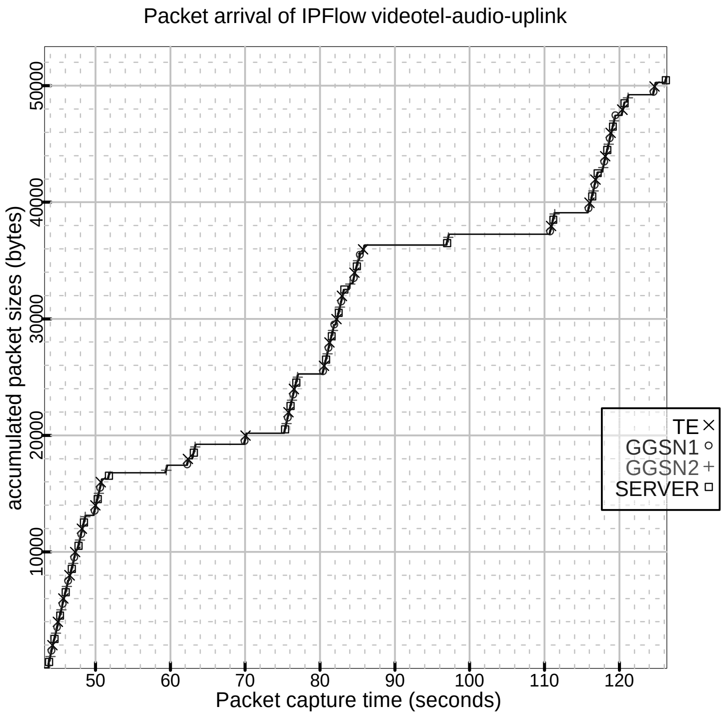
4.2.2.1 Audio flow, downlink

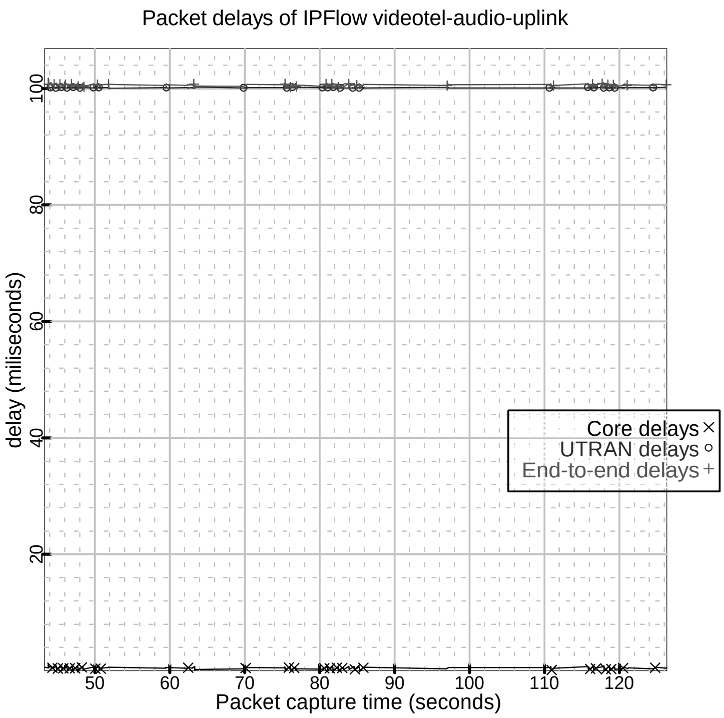




The two plots above refer to the downlink audio stream of video telephony. Once again, we witness the effect of the on-off nature of natural speech in the service curves. Although not perceptible on the service curves, there is a delay of ~100ms in the UTRAN, as can be observed in the delay plot. It is also clear that the core network delay is negligible - a clear indication that the RSVP reshaping is not introducing any additional delay.

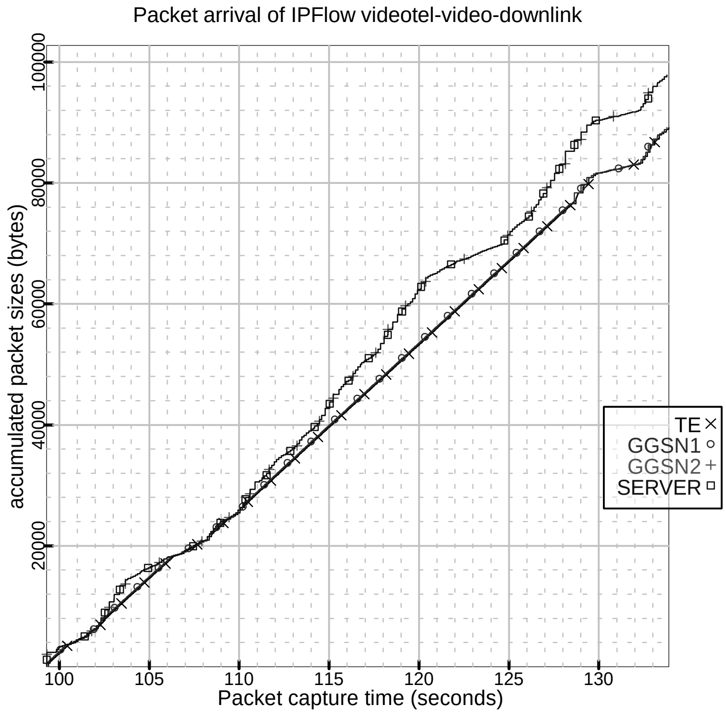
4.2.2.2 Audio flow, uplink

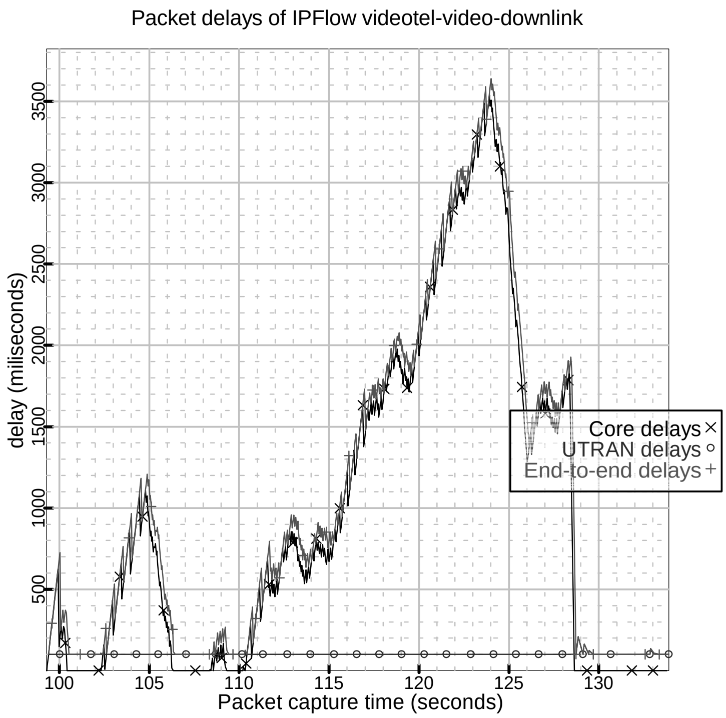




The two plots above refer to the uplink audio stream of video telephony. The same comments as for the downlink case apply here, as well.

4.2.2.3 Video flow, downlink



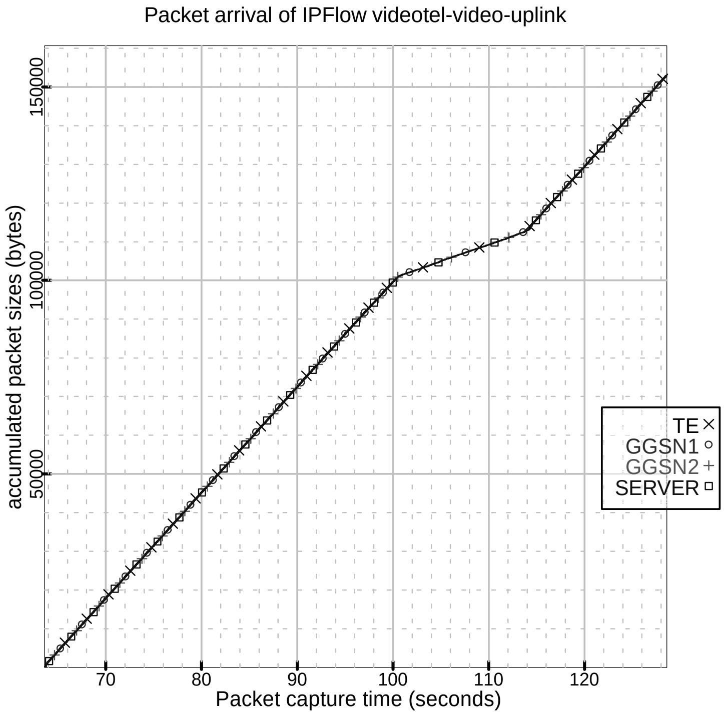


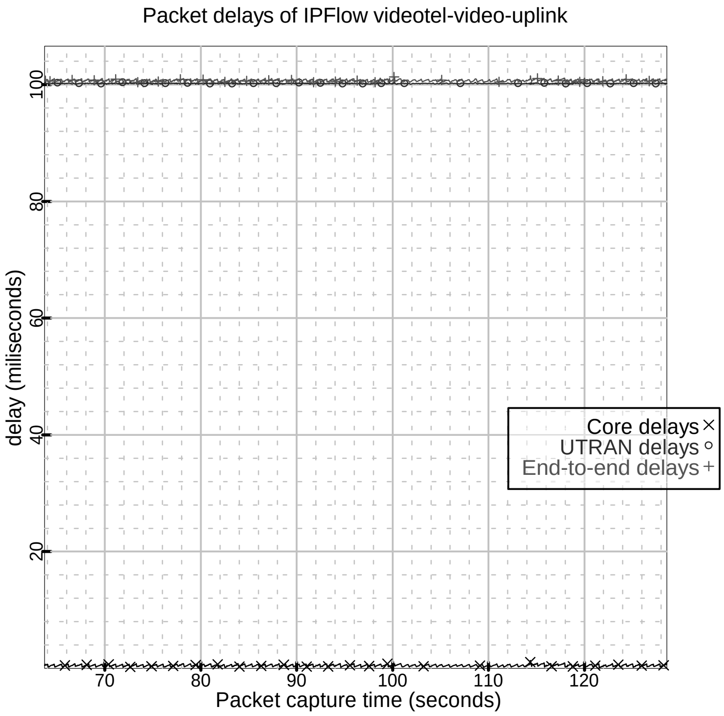
The two plots above refer to the downlink video stream of video telephony. The service curves are very “interesting”. They show that the application was, once again, generating a much larger bit rate than the RSVP reservation. This is an application configuration issue.

The service curves also indicate that there were heavy packet losses. We can conclude this by looking at the end of the curves and asserting that there is a disparity in the final byte count of SERVER/GGSN2 versus GGSN1/TE, which is also an indication that the losses occurred between interfaces GGSN2 and GGSN1, i.e., at the reshaping point. What happened was that the TBF module, which takes care of reshaping, has a large but limited buffer. In this case the buffer has overflowed.

Comparing with the delay plot, we can reassure our prediction. It is clear that core network delays grow steadily, accompanying the TBF buffer growth. At some point, some packets are discarded, and the delays drop to zero. Once again, the 100 ms UTRAN delay is very clear, which indicates the proper functioning of NistNet. The problem can be eliminated by reducing the rate generated by the VIC application. This case was presented just to demonstrate the ability of these plots to detect potential problems.

4.2.2.4 Video flow, uplink





The two plots above refer to the uplink video stream of video telephony. Here, we cannot see the reshaping in this instance because the testing framework does not cover it. We see only traffic after reshaping, which explains why it looks so "well shaped".

4.3 Audio – video streaming

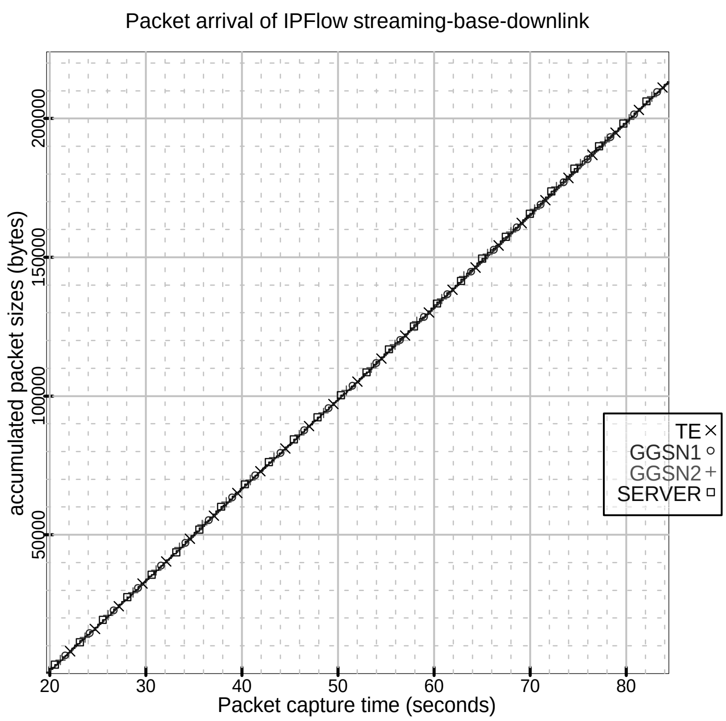
This service allows a user located at the terminal equipment to retrieve a film stored in the server. The film is sent as two unidirectional IP flows (one for audio and one for video). Each flow is transported in a dedicated streaming and unidirectional Secondary PDP Context. The video flow can be renegotiated and is well adapted to a pre-defined set of transport bit rates.

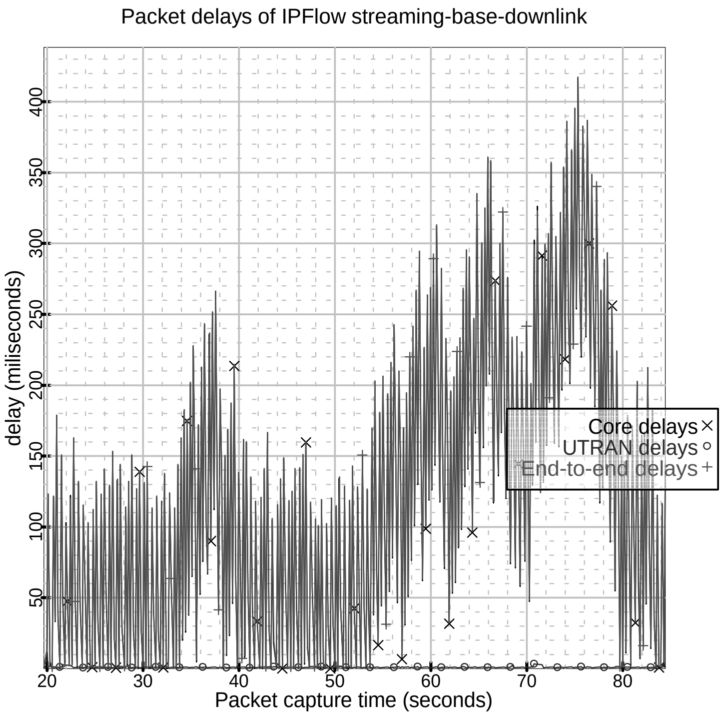
Audio-video streaming is based on the applications developed by MPEG4IP project [28]. The audio base layer stream is delivered over UDP and carried in one PDP Context from the Server to the UE. The video layer stream is delivered over UDP and carried in another PDP Context from the Server to the UE.

4.3.1 Without UTRAN

DSS and gmp4player run QoS reservation, which means they are used with the QoS Manager. NistNet is not used. Therefore, PC#2 becomes a simple IP router using only Ethernet layer 2 technology (100 Mbit/s). Audio stream (32 kbit/s) corresponds to the base layer, while the Video stream (up to 96 kbit/s) corresponds to the enhancement layer.

4.3.1.1 Audio flow, downlink

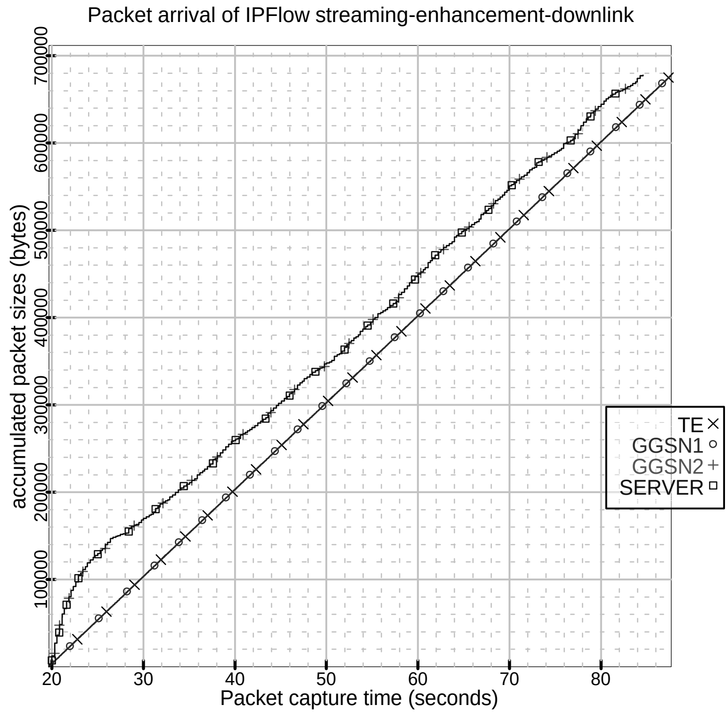


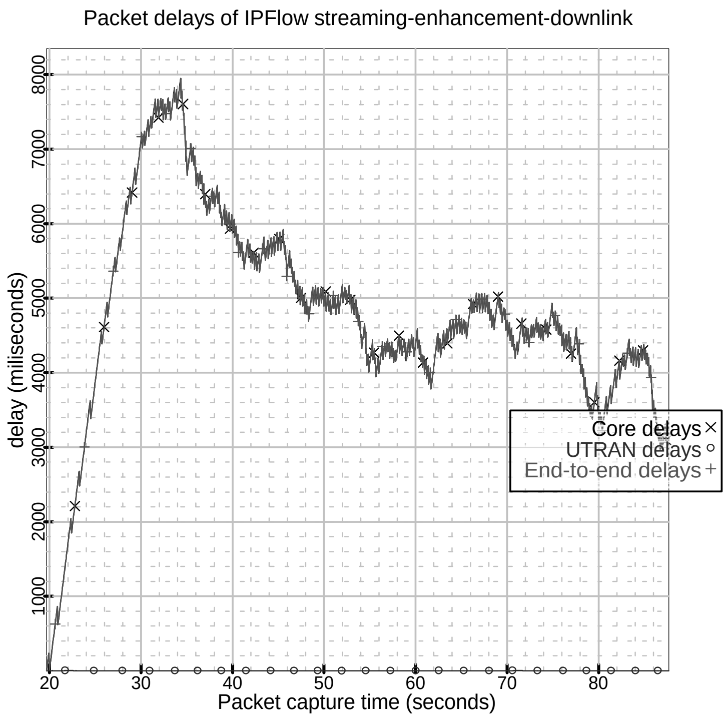


The two plots above refer to the audio stream, which flows in the downlink direction. For that reason, the plots show the effect of the RSVP reshaping. Looking at the service curves, the audio flow looks well shaped even before reshaping. Core delays, which also represent the effect of shaping at GGSN, look strange. Its mean value clearly represents the shaping effect. The sharpest, highly variable values are more difficult to explain.

We know that audio is composed of a large number of very small packets. The reason why this has such an important impact on delays is that, with more packets, the number of collisions at the Ethernet level is higher. Collisions at the HUB used to interconnect the PCs on lab may cause increased delays because of retransmissions, and are very unpredictable. The end result is a very “noisy” delay plot. An alternative explanation for this would be that there is a time synchronization program running on each node with a period of, in this case, one second.

4.3.1.2 Video flow, downlink



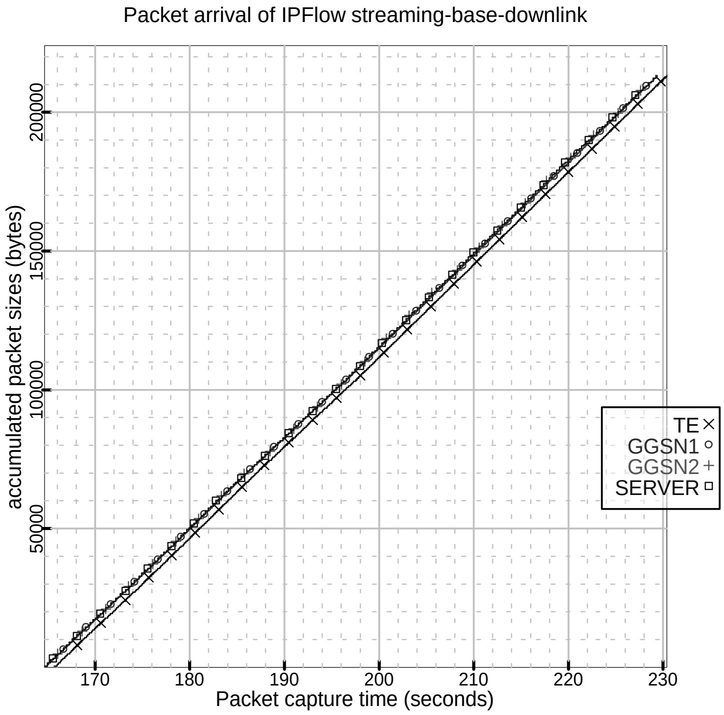


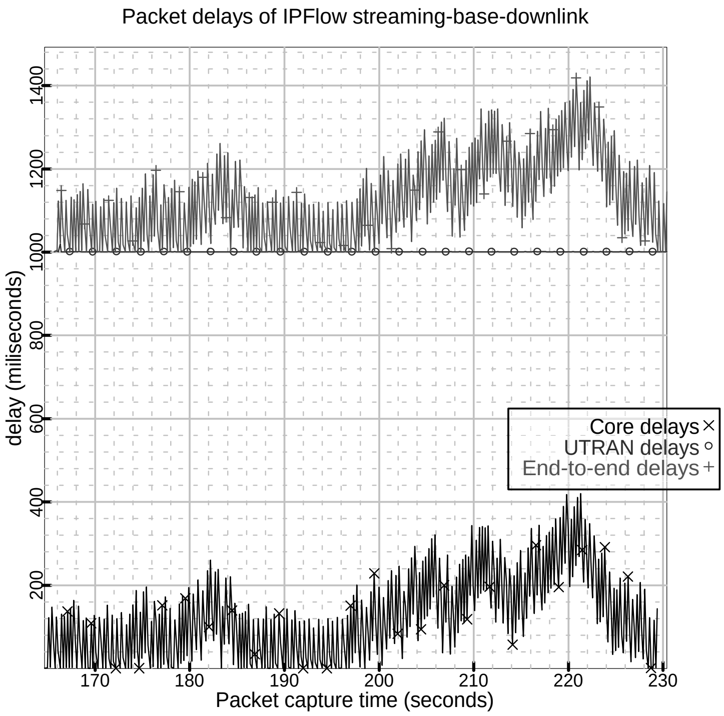
The two plots above refer to the video stream. The traffic pattern depicted in the service curves plot is very typical of video streaming. There is an initial peak rate, when the film starts, with a corresponding increased delay (7 to 8 seconds in this case) followed by a decrease in bit rate and a gradually decreasing delay. The reshaping point is the one responsible for the bulk of end-to-end delay, as expected. It is important to point out that, although the maximum delay of 8 seconds looks bad, it really is acceptable. This delay only affects the time it takes for the stream to start displaying after being requested; otherwise it is not visible.

4.3.2 With UTRAN emulation

The purpose of this demonstration is to show the impact of PDP Contexts into the end-to-end video streaming service. DSS and gmp4player multimedia applications run with QoS reservation, which means they are started with the testbed running and register with the QoS Manager. NistNet was configured to emulate the audio and video PDP contexts: a) audio - bandwidth=32kbit/s, delay=1s; b) video - bandwidth=96kbit/s, delay=1s.

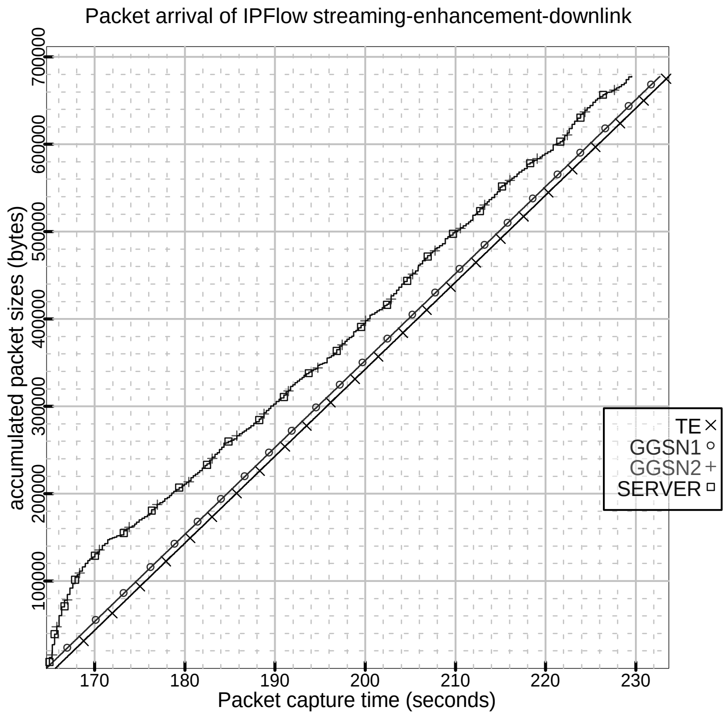
4.3.2.1 Audio flow, downlink

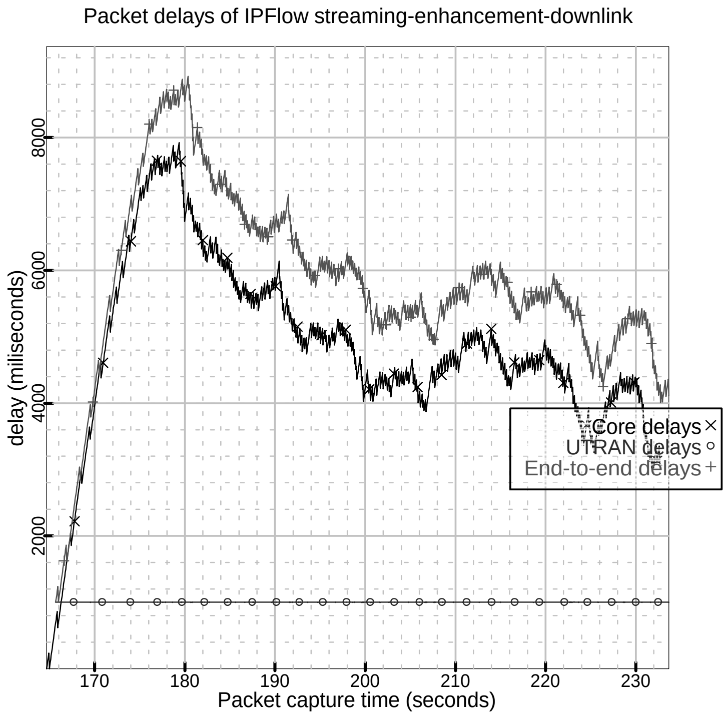




The two plots above refer to the audio stream. The 1s delay in the UTRAN emulator is visible in both plots: as a horizontal right shift of the TE curve in the first plot, and directly as the horizontal line at 1000 ms in the UTRAN delay curve, in the second plot. The core delay curve is an almost exact replica of the previous trial, if the different vertical scale is accounted for. Especially important is the fact that the UTRAN delay curve is a horizontal line. This means that the traffic after reshaping does not exceed the bandwidth limits configured in NistNet; otherwise we would see an additional and variable delay in the UTRAN delay curve.

4.3.2.2 Video flow, downlink





The two plots above refer to the video stream. The video stream has a very high bit rate in the beginning, but stabilizes to a lower average value after some time. The UTRAN delays are steady at one second, which indicates, once again, that the bandwidth limit is not exceeded.

5 CONCLUSIONS

This document describes the QoS framework developed under the ARROWS project. This QoS framework provides a solution for deploying IP based multimedia applications over UMTS access networks and provides these applications with the Quality of Service they require. The solution assumes that IntServ/RSVP is supported in the UMTS access network that, in turn, can interoperate with DiffServ IP core networks.

The key contributions of our work are presented next.

The gap closed. We claim that we have closed the gap between the IP IntServ model and UMTS QoS in a simple and effective way. This consists basically in allowing the uncommon layer 2 technology offered by UMTS to be used by IntServ/RSVP. Layer 2 circuits (UMTS bearers) cannot be opened from both sides (TE and GGSN) and transmission characteristics vary with time. Our solution solves these problems and, by doing it, we allow standard IP QoS solutions to be deployed over UMTS networks.

The QoS Manager. This functional block, used in mobile terminals, is the entity that closes the gap referred to above. It behaves like the mobile end point of RSVP but it is also responsible for aggregating IP flows into PDP Contexts and by mapping IP QoS parameters into UMTS QoS parameters. Moreover, but not less important, it provides a simple API that allows standard and non-RSVP aware applications to be easily deployed in UMTS mobile terminals.

The role of PDP Contexts. The role of PDP contexts has also become clear. The Primary PDP Context is used to support the standard IP best effort traffic. Secondary PDP contexts are left for special QoS demanding applications; conversational and streaming services will be those more likely to be used.

6 REFERENCES

[1] WP3, “D05 – Testbed and Multimedia Applications Specification”, ARROWS project.
[2] WP3, “D10 – Implemented Multimedia Applications”, ARROWS project.
[3] WP3, “D15 – Applications integration on the testbed”, ARROWS project.
[4] 3GPP TS 24.007: “Mobile radio interface signalling layer3; General aspects”.
[5] 3GPP TS 23.107 v3.6.0: “QoS Concept and Architecture”.
[6] RFC 2205 “Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification”, September 1997.
[7] “RAPI - An RSVP Application Programming Interface Version 5”, Internet Draft, http://www.isi.edu/rsvp/DOCUMENTS/rsvpapi.txt.
[8] 3GPP TS 23.207 V5.0.0: “End-to-End QoS Concept and Architecture”.
[9] 3GPP TS 24.008 V5.0.0 (2001-06), “Mobile radio interface layer 3 specification Core Network Protocols – Stage 3”.
[10] Sally Floyd, "Notes on CBQ and Guaranteed Service", 1995.
[11] IP QoS Page, http://qos.ittc.ukans.edu/
[12] ISI RSVP Project Page, http://www.isi.edu/div7/rsvp/rsvp.html
[13] Xipeng Xiao, Lionel Ni, “Internet QoS: A Big Picture”, *IEEE Network Magazine*, Vol. 13, No. 2, pp. 8-18, March/April 1999.
[14] K. Dovrolis, M. Vedarn, P. Ramanathan, “The Selection of the Token Bucket Parameters in the IETF Guaranteed Service Class”, Technical Report, Department of ECE, University of Wisconsin, November 1997.
[15] RFC 2998 “A Framework for Integrated Services Operation over DiffServ Networks”, November 2000.
[16] RFC 2543 “SIP – Session Initiation Protocol”, March 1999.
[17] RFC 2974 “SAP – Session Announcement Protocol”, October 2000.
[18] RFC 2327 “SDP – Session Description Protocol”, April 1998.
[19] RFC 1889 “RTP – Real-time Transport Protocol”, January 1996.
[20] RFC 2326 “RTSP – Real-time Transport Streaming Protocol”, April 1998.
[21] RFC 2210 “The Use of RSVP with IETF Integrated Services”, September 1997.
[22] RFC 2215 “General Characterization Parameters for Integrated Service Network Elements”, September 1997.
[23] RFC 2212 “Specification of Guaranteed Quality of Service”, September 1997.
[24] RFC 2209 “Resource ReSerVation Protocol (RSVP) - Version 1 Message Processing Rules “, September 1997.
[25] RFC 2208 “Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement Some Guidelines on Deployment “, September 1997.
[26] Vic homepage: http://www-mice.cs.ucl.ac.uk/multimedia/software/vic/
[27] RAT homepage: http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/
[28] Mpeg4ip homepage: (http://mpeg4ip.sourceforge.net/)
[29] Nistnet homepage: http://snad.ncsl.nist.gov/itg/nistnet/