Uppgjord (även faktaansvarig om annan) - *Prepared (also subject responsible if other)*		Nr - *No.*		
EMN/K/A Georgios Karagiannis (5370), EMN/K/A Vlora Rexhepi (5382)		9/036—FCPNB 10288 Uen		
Dokansv/Godk - *Doc respons/Approved*	Kontr - *Checked*	Datum - *Date*	Rev	File
EMN/K/A Geert Heijenk (5430)		2000-09-06	A	

A Framework for QoS & Mobility in the Internet Next Generation

Abstract

It is expected that the next generation Internet architecture will support applications with different quality of service requirements, independently of whether their location is fixed or movable. However, enabling QoS in Internet is a tough challenge, and it gets even tougher when the mobile environment with its non-predictive characteristics is introduced. In this paper we propose a framework that will integrate various QoS architectures and mobility protocols and will offer the freedom to users to choose between different wireless and wired access technologies based on certain predefined criteria, e.g. based on QoS parameters.

Uppgjord (även faktaansvarig om annan) - *Prepared (also subject responsible if other)*
EMN/K/A Georgios Karagiannis (5370),
EMN/K/A Vlora Rexhepi (5382)

Nr - *No.*
9/036—FCPNB 10288 Uen

Dokansv/Godk - *Doc respons/Approved* | Kontr - *Checked*
EMN/K/A Geert Heijenk (5430)

Datum - *Date* | Rev | File
2000-09-06 | A

Contents

1 Introduction

The diversity of the current Internet applications from the most simple ones like e-mailing and web browsing going to high demanding real time applications like the IP telephony and multimedia conferencing, has raised the expectations that both, users and software developers of these applications have from Internet. On the other hand, in such a highly competitive environment as the Internet Service Providers (ISPs) world, satisfying customer needs, regardless of whether they are other ISPs or end users is key to their survival. Therefore the ISPs zeal to provide value-added services to their customers is growing immensely. These demands have led to evolvement of QoS on Internet as a necessity. Enabling QoS on the best effort Internet model introduces complexity in several aspects, starting from applications, different networking layers and network architectures but also in network management and business models. All these aspects have been major research topics over the past few years. Finding an efficient solution for end-to-end QoS over Internet i.e. the IP networks that will satisfy both ISPs and their customers is a tough venture.

It becomes even tougher when one is introducing QoS in an environment of mobile hosts, wireless networks, and different access technologies, because of wireless networks dynamically changing topologies and resources. Yet, the need for QoS mechanisms in this environment is greater due to scarce resources, unpredictable available bandwidth and variable error rates.

The rapid growth of mobile systems indicates that the future Internet will have to deal with mobile users that will use the same diversity of applications as fixed users. Thus, solutions for enabling QoS over IP should take into account mobility issues also, in order to be able to fulfil these upcoming requirements of future Internet users.

The current work on the QoS over IP architectures, i.e. Integrated Services and Differentiated Services seems to leave out the mobility support, despite its importance. Therefore, in this report we introduce a framework for QoS and Mobility in the next generation Internet that will integrate the existing QoS over IP architectures and protocols supporting mobility. We have also introduced new entities necessary for the entire framework functionality. Section 2 gives an overview of the current IP QoS architectures and mechanisms, IP mobility and session protocols. A general introduction of the framework, its entities and protocols is given in Section 3, where we also describe and identify several QoS mobility service classes. The proposed integration and interoperability of the protocols given in Section 2 and the possibility of their interworking is described in Section 4. Section 5 describes the framework architecture operation. Finally the conclusions and acknowledgements are given in sections 6 and 7, respectively.

Note that the ideas proposed in this document by the authors do not imply any kind of Ericsson strategy.

2 Current IP QoS Architectures, IP Mobility Protocols and Session Protocols

The framework we propose relies on interworking between the current IP QoS architectures, i.e. Integrated Services and Differentiated Services, the IP mobility protocols and session protocols. In this section we give an overview of the current IP QoS architectures and mechanism, protocols for IP mobility support and session protocols.

Section 2.1 and the following subsection gives an overview of IP QoS architectures and also of a number of mechanisms that are proposed to improve the flexibility of IP QoS architectures and enable interoperability between them, when it comes to their wider deployment in the Internet. Section 2.2 explains shortly the Session Initiation Protocol (SIP) as a representative of protocols for negotiating, managing and controlling the sessions. Section 2.3 and its following subsections give a description of the protocols supporting IP mobility.

Note that in this section we only mention those existing architectures, mechanisms and protocols that at this moment we consider relevant to our framework. The proposal on their integration and interoperability is explained later in this report, i.e. section 4.

2.1 IP QoS Architectures and Mechanisms

The efforts to enable end-to-end QoS over IP networks have led to the development of two different architectures, the Integrated Services architecture and more recently, the Differentiated Services architecture.

2.1.1 Integrated Services Architecture

The Integrated Services (Intserv) architecture [RFC1633] uses an explicit mechanism to signal per-flow QoS requirements to network elements (hosts, routers). Network elements, depending on the available resources, implement one of the defined Intserv services (Guaranteed or Control Load service) based on which QoS will be delivered in the data transmission path. The RSVP signaling protocol [RFC2205], [RFC2210] was designed as a dynamic mechanism for explicit reservation of resources in Intserv, although Intserv can use other mechanisms as well. It is initiated by an application at the beginning of a communication session. But, even though Intserv is designed to provide end-to-end QoS it is currently not widely deployed. As it is emphasized so many times by now, due to maintenance and control of per-flow states and classification, reserving resources per-flow introduces severe scalability problems at the core networks, where the number of processed flows is in a millions range. Consequently the usage of the Integrated Services architecture is limited to small access networks where the number of flows using reservations is modest.

The simplified RSVP/Intserv framework is shown in Figure 2-1. As it is shown every RSVP aware router in the Intserv will be able to perform RSVP signalling, admission control, policing and scheduling.

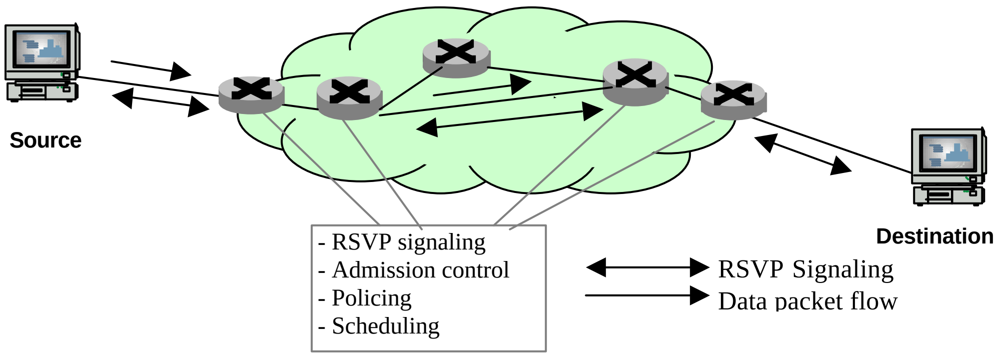


Figure 2-1: RSVP/Intserv framework

2.1.2 RSVP

The Resource Reservation Protocol (RSVP) (see [RFC2210], [DuYa99]) is a signalling protocol that can be used by an application to inform its QoS requirements to an Internet infrastructure. RSVP is initiated by an application at the beginning of a communication session. A communication session is identified by the combination of the IP destination address, transport layer protocol type and the destination port number.

The resources reserved by the RSVP for a certain communication session will be used for all packets belonging to that particular session. Therefore, all RSVP packets will include details of the session they belong. The main RSVP messages are the PATH and RESV messages. The PATH message is sent by a source that initiates the communication session and it explicitly binds the data path of a flow. Furthermore, it describes the capabilities of the source. The RESV message is issued by the receiver of the communication session and it follows exactly the path that the RSVP PATH message has followed hop by hop back to the communication session source. The RESV message in its way back to the source it may install QoS states at each hop. These states are associated with the specific QoS resource requirements of the destination. The RSVP reservation states are temporary states, i.e., soft states, that have to be updated regularly. This means that PATH and RESV messages will have to be periodically retransmitted. If these states are not refreshed then they will be removed.

The RSVP protocol uses additional messages that are used to either provide information about the QoS state or to explicitly delete the QoS states along the communication session path.

The RSVP messages and their meaning are given in Table 2-1.

RSVP Messages	
RSVP Message Name	**RSVP Message Function**
PATH	The PATH message is sent by a source that initiates the communication session and it explicitly binds the data path of a flow. Furthermore, it describes the capabilities of the source.
RESV	The RESV message is issued by the receiver of the communication session and it follows exactly the path that the RSVP PATH message has followed hop by hop back to the communication session source. The RESV message in its way back to the source it may install QoS states at each hop. These states are associated with the specific QoS resource requirements of the destination. The RSVP reservation states are temporary states, i.e., soft states, that have to be updated regularly
PATH Error	Is used to report errors that are occurring during the installation of a path from the source to the destination of a communication session.
RESV Error	Is used to report errors that are occurring during the installation of the reservation states along the communication session path.
RESV Confirm	It provides a positive indication to the initiator of the communication session informing that all nodes along the communication session path accepted the reservation request. The RSVP Confirmation messages are typically sent by the source of the communication session directly to the destination of this communication session. Intermediate nodes do not process RSVP confirmation messages.
PATH Tear	Is sent by the source of the communication session and it explicitly deletes the stored QoS state information on all nodes included in a communication session path.
RESV Tear	Is sent by the destination of the communication session and it explicitly deletes the stored QoS state information on all nodes included in a communication session path.

Table 2-1: The RSVP messages

2.1.3 Differentiated Services Architecture

The Differentiated Services (Diffserv) architecture [RFC2475], [RFC2638], [BeBi99] was introduced as a result of the efforts to avoid the scalability and complexity problems of Intserv. Per-flow state is pushed to the edges and the traffic through Diffserv routers is treated on aggregate basis. The service differentiation is achieved by means of Differentiated Service (DS) field in the IP header and the Per-Hop Behaviour (PHB) as main building blocks. At each node packets are handled according to the PHB invoked by the DS byte in the packet header. The PHB defines the externally observable behaviour at the node. Two PHBs have been defined, the assured forwarding (AF-) PHB [RFC2597] and the expedited forwarding (EF-) PHB [RFC2598]. The Diffserv domain will provide to its customer, which is a host or another domain, the required service by complying fully with the agreed Service Level Agreement (SLA). SLA is a bilateral agreement between the boundary domains negotiated either statically or dynamically. The transit service to be provided with accompanying parameters like transmit capacity, burst size and peak rate, is specified in the technical part of the SLA, i.e. the Service Level Specification (SLS). The Diffserv architecture is certainly promising, but there are a lot of open issues related to intra-domain resource allocation mechanisms and inter-domain communication in case of dynamic resource provisioning that need to be defined and researched. The simplified Diffserv framework is given in Figure 2-2.

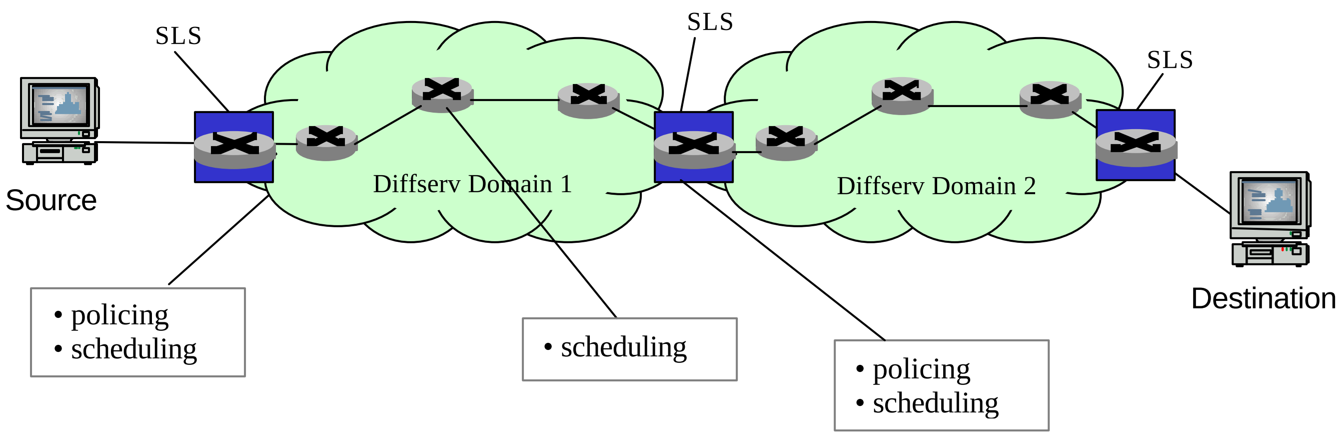


Figure 2-2: Differentiated Services framework

2.1.4 RSVP Aggregation

The RSVP aggregation (see [Balt00]) concept is used to reduce the Intserv scalability problems by extending the RSVP protocol with facilities for aggregation of individual reserved sessions into a common class and across transit domains. It describes mechanisms for dynamic creation of the aggregate reservation, classification of the traffic for which the aggregate reservation applies, determination of the bandwidth needed to achieve the requirement and recovery of the bandwidth when the sub-reservations are no longer required. A RSVP aggregated session is identified by the IP destination address, the protocol ID and the DSCP information. Furthermore, for classification and scheduling of traffic supported by aggregate reservations Diffserv mechanisms are used. Diffserv DSCPs are used to identify traffic covered by aggregate reservations and one or more Diffserv PHB's are used to offer the required forwarding treatment to this traffic.

The first router in the transit domain that handles the aggregated reservations is called Aggregator, while the last router in the transit domain that handles the reservations is called Deaggregator. An RSVP aggregation region is consisted by routers that are capable of managing the RSVP aggregated states.

The RSVP aggregation concept can be used either when RSVP is applied end to end or edge to edge. In the later case the Aggregator can use a policy that can be based on local configurations and local QoS management architectures, to set the DSCP packets that are passing into the aggregated region. For example, the Aggregator may be a PSTN (Public Switched Telephone Network) gateway that aggregates a set of incoming calls and makes an aggregate reservation across one or more Diffserv domains up to the Deaggregator that can be e.g., another PSTN gateway. In this situation the call signalling is used to establish the E2E reservations.

In the following text we describe the situation that the RSVP aggregation concept is used when RSVP is applied end to end.

Each end point can initiate a per flow and end to end RSVP resource reservation procedure. The per-flow RSVP resource reservation messages are referred as E2E PATH/RESV and the aggregated resource reservations messages are referred as aggregated PATH/RESV. The E2E PATH/RESV messages are carried though the RSVP aggregation region transparently, since their RSVP protocol Id at the aggregator will be replaced with a new protocol number, i.e., RSVP-E2E-Ignore that is expected to be standardised. Each E2E RSVP resource reservation request that arrive at the ingress router, i.e., Aggregator, of an RSVP aggregated region can initiate or resize a RSVP aggregated reservation. In the RSVP aggregated region the aggregated RSVP messages are managing the reservation of the resources for a set of per-flow RSVP reservations. The E2E RSVP reservation states are temporary states, i.e. soft states that have to be updated regularly. This means that E2E PATH and E2E RESV messages will have to be periodically retransmitted. If these states are not refreshed then they will be removed. These states may also be removed by using the E2E PATHTear and E2E RESVTear messages. When these E2E states are removed then their FLOWSPEC information must be removed from the allocated portion of the aggregate reservation, such that this same bandwidth will be re-used for other traffic in the near future. Furthermore, their SENDER_TSPEC information must also be removed from the aggregated state.

It is important to note that in [BaIt00] it is suggested that a predefined policy should exist to maintain the amount of bandwidth required on a given RSVP aggregated reservation that is taking into account the sum of its underlying E2E reservations, but that will protect it from changing it frequently. This can be achieved by using some level of trend analysis.

From the above information we can deduce that based on a certain policy the Aggregator and Deaggregator will decide when the RSVP Aggregated states will be refreshed or updated and therefore this triggering time is not completely defined by the E2E RSVP messages.

Within an aggregation region three possible scenarios can be distinguished.:

- **Signalling Flow for the first E2E flow**

The flow diagram depicted in Figure 2-3 illustrates a detailed flow of RSVP messages in the case when there is no Aggregated PATH between the Aggregator and Deaggregator. The number of the Aggregated PATH states that will have to be installed in each router depends on the number of the supported DSCPs. In Figure 2-3 it is considered that two DSCPs are supported, e.g., EF and AF PHB's, and therefore, two RSVP PATH aggregated states have to be created. The symbols (A) and (B) in the flow diagram represent new aggregation values needed for the different supported DCSP's. The E2E RSVP messages transport the value that identifies a particular DSCP (e.g., PHB) type in the DCLASS object (see [Balt00]).

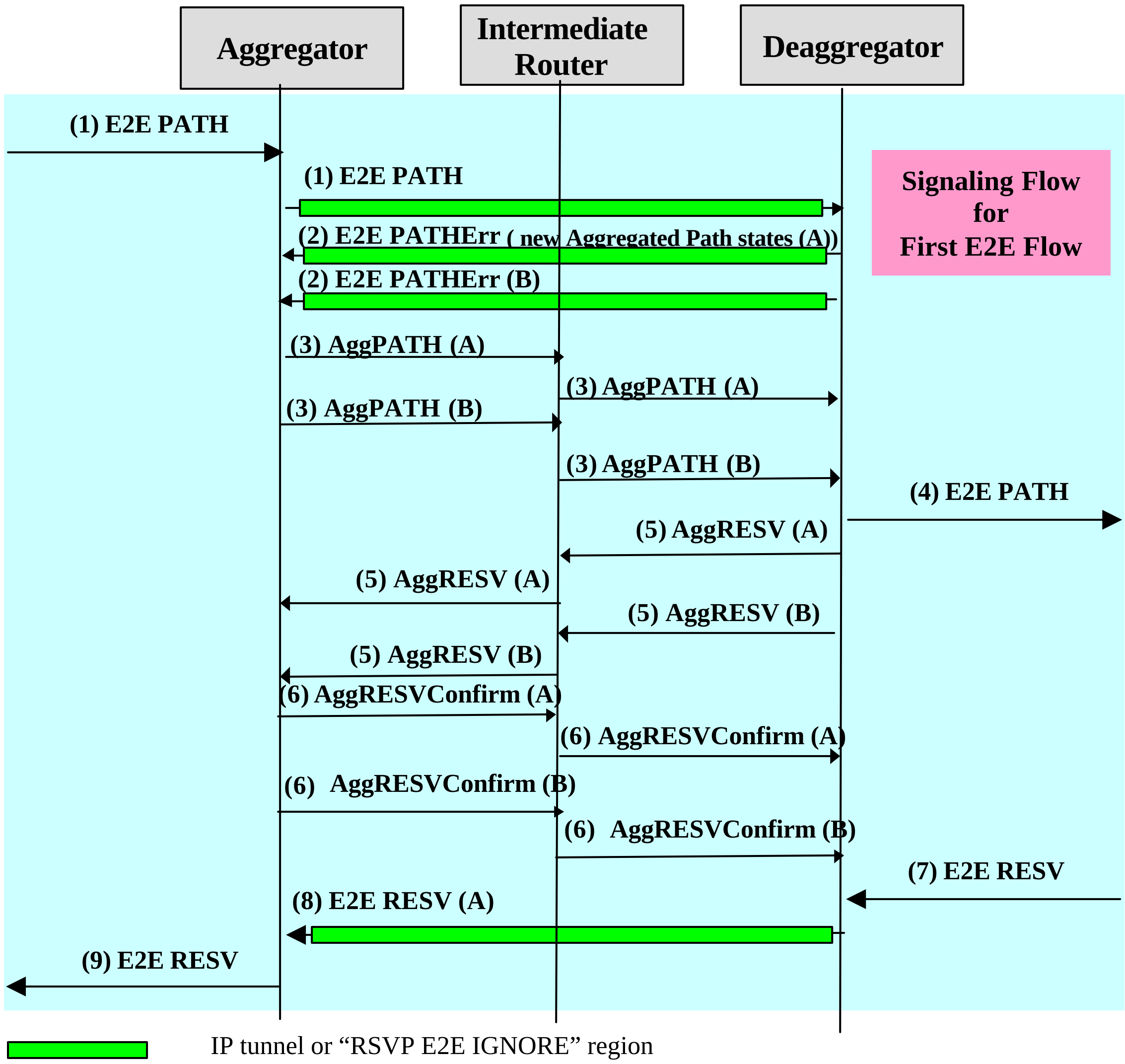


Figure 2-3: RSVP aggregation signaling flow for first E2E flow

- **Signalling Flow for subsequent E2E Flow without reservation resizing**

Figure 2-4 illustrates a detailed flow of RSVP messages in the case that there already exist an Aggregated PATH between the Aggregator and Deaggregator and there is no need for a change in the RSVP aggregated reservation. The E2E RSVP messages transport the value that identifies a particular DSCP (e.g., PHB) type in the DCLASS object (see [Balt00]).

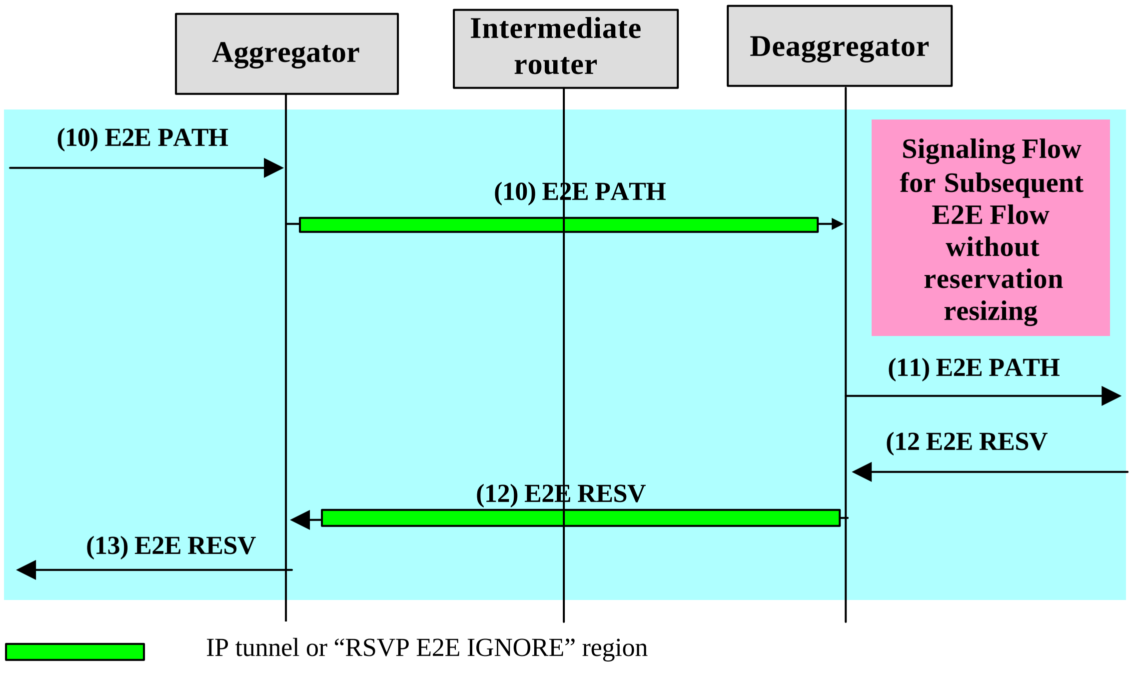


Figure 2-4: RSVP aggregation signaling flow for subsequent E2E flow without reservation resizing

- **Signalling Flow for subsequent E2E Flow with reservation resizing**

Figure 2-5 illustrates a detailed flow of RSVP messages in the case when there already exist an Aggregated PATH between the Aggregator and Deaggregator and there is a need for a change in the RSVP aggregated reservation (C – represents new values, e.g. more bandwidth). The E2E RSVP messages transport the value that identifies a particular DSCP (e.g., PHB) type in the DCLASS object (see [Balt00]).

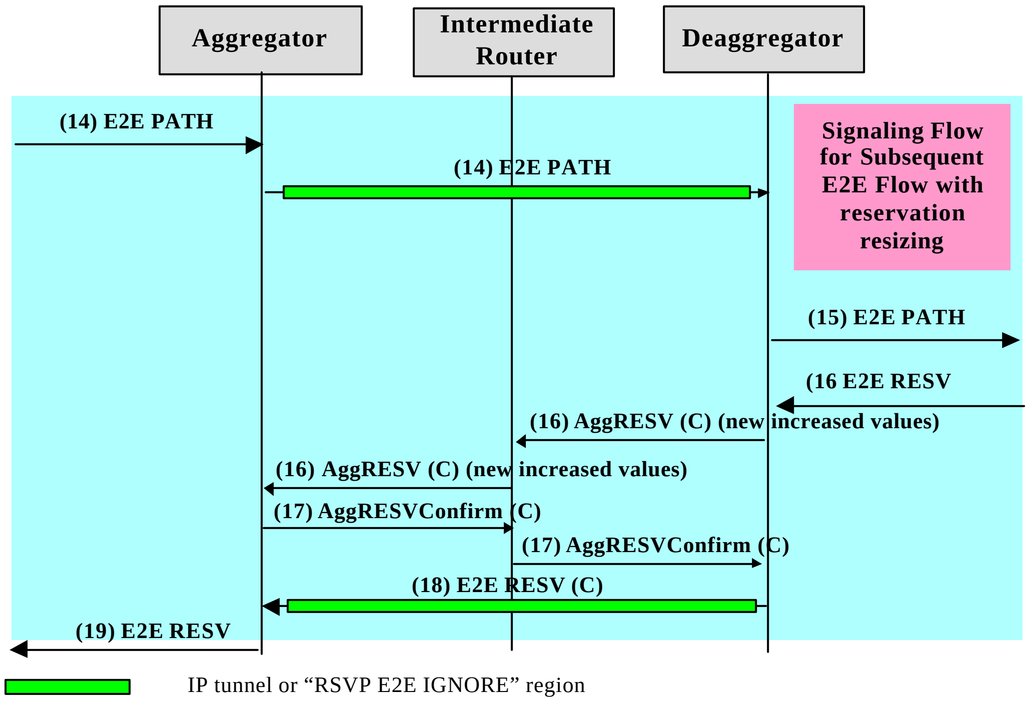


Figure 2-5: RSVP aggregation signaling flow for subsequent E2E flow with reservation resizing

- **Signalling Flow for E2E Flow release**

The flow diagram depicted in Figure 2-6 illustrates an E2E RSVP release procedure.

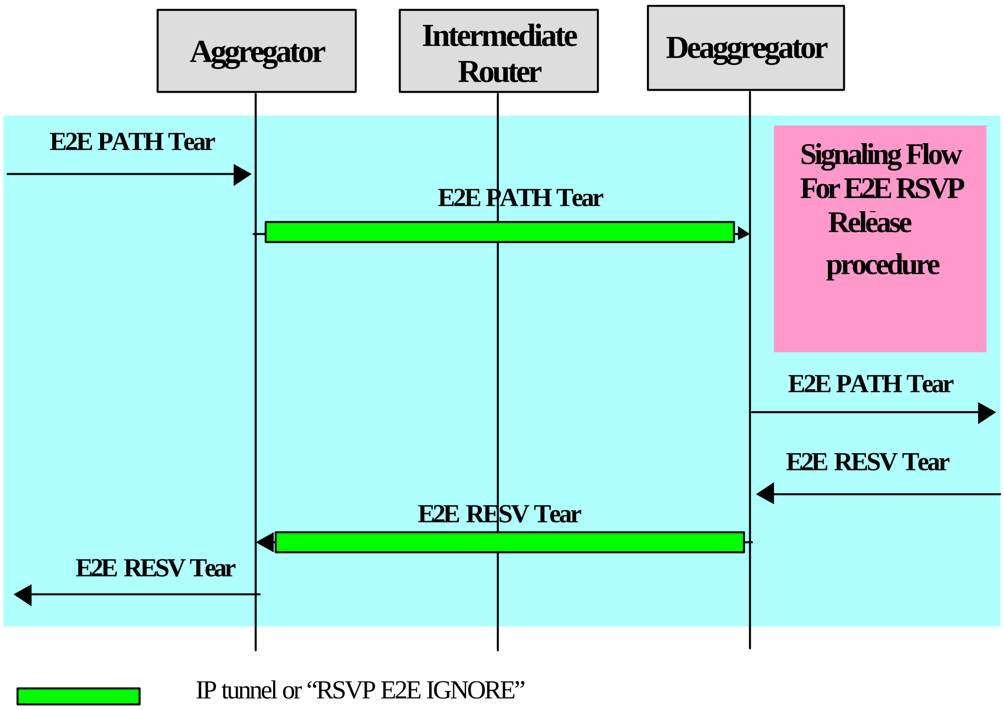


Figure 2-6: RSVP aggregation signaling flow for E2E release

2.1.5 Load Control

Load control is a scheme for resource allocation within the Diffserv networks [WeTu00], without requiring explicit signalling or any per-flow processing in core routers. The load control scheme is designed such as to perform admission control of incoming request and to drop the admitted flows in case of failure events, e.g. link failure. The Load Control related information can be stored in the Diffserv packet headers by using either new Differentiated Services Code Points (DSCP) or using the two least significant bits of either IPv4 TOS (Type of Service) octet or the IPv6 Traffic class octet. The Load control scheme has two modes of operation:

- **Simple marking**

The simple marking is a measurement – based admission scheme where the routers measure the traffic volume and base the marking on these result. The operation of this scheme can be accomplished in the following steps:

- Resource Probing: the initiating edge device sends a (PP) Probe Packet (see explanation below) into the network. This packet will pass from all routers as the normal IP traffic will pass through. The marking function of the router will perform an OR operation of its own (Load) status and the status of the PP packet. The PP packet will arrive at the egress router. The load status of this packet will reflect the aggregated resource status along that path.

- Send resource status to the ingress: The egress edge device after receiving the PP packet has to copy the marker from the header of the received packet to the header/payload of a reverse packet and sends it back to the initiating edge device. The PP packet may be discarded or changed to an ordinary packet or encapsulated and sent back to the ingress edge router. In some situations the PP packet can be used for checking bi-directional resources.

- Acceptance/Rejection: The ingress edge router after receiving the packet sent by the egress router, it will be notified about the load status of the network and it will admit or block the request by setting up appropriate packet filtering, measuring and marking rules.

- Reaction to exceptional events: In case of sever congestion on an interface, the router changes the load status of all packets that are passing through that interface into (MP) Marked Packets.

The packets used in this scheme are the PP, MP and Ordinary Packets (OP).

- Unit-based reservation

This scheme can be used to perform resource reservation of the amount of traffic that is transmitted by allowing the sources to keep their reservations independently of the volume of the actual traffic transmitted. In this scheme the router uses certain packet types (see below PP and RP) to request and maintain the reserved resources for flows. Each flow can occupy a certain number of units of resources, which are a share of bandwidth that can be reserved by the edge devices.

Some characteristics of the unit-based reservation are:

- number of units per one refreshment period specifies the amount of resources to be reserved

- each micro-flow occupies any number of units per refreshment period

- reservations are timed out after a refreshment period (soft state). The length of this period is uniform in the Diffserv domain

The packets used in this scheme are:

- OP (Ordinary Packet): ordinary IP data packet

- PP (Probe Packet): used during a current refreshment period (say period (*i*)) by all routers in the Diffserv network that received the PP message to reserve one unit of resource. This will be applied during the next refreshment period (*i+1*); flows requiring more bandwidth send more PP packets per refreshment period; Note that one unit of resource is reserved only if the PP does not become a marked packed (see below);

- RP (Refreshment Packet): used during the subsequent refreshment period (*i+1*) to refresh the reservations that were done by the PP packets during the period (*i*); if they passed unmarked then each RP packet reserves one unit of resource that will be applied by the Diffserv network during the period(*i+2*);

- MP (Marked Packet): During transition in network a PP or RP can become MP, meaning that at least one router reject the reservation. Moreover, during transition in network a OP can become MP meaning severe congestion on a interface;

Figure 2-7 views the basic operation of the Load control unit based-reservation scheme. The resource reservation, during one refreshment period (i.e. period (i)), can be achieved by sending PP or RP messages. If a router changes the PP or RP messages into an MP packet, it means that the resource reservation procedure for that unit of resource could not be accomplished. If these messages are not changed in MP messages then it means that the resource reservation procedure has been able to reserve the resources for that unit of resource during period (i+1). The resource release procedure during a period can be achieved when the resource reservation mechanism does not send any PP or RP messages, but only OP messages.

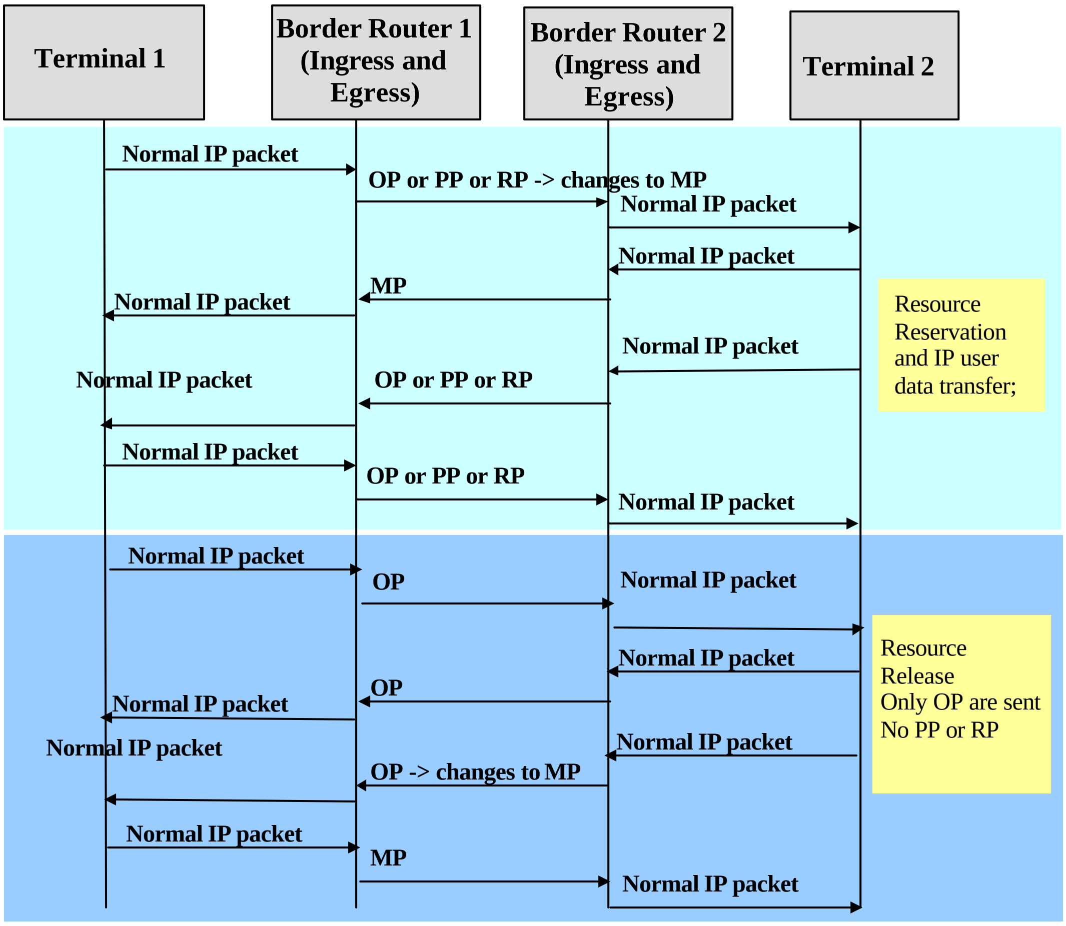


Figure 2-7: Resource reservation and resource release procedures in Load Control

In Figure 2-7 the Load Control operation is accomplished by providing the possibility to the Ingress/Egress Border Routers to use the Normal IP packets that are sent by the end Terminals and to mark them as PP, RP, OP or MP packets.

However, there are situations that the Ingress/Egress Border Routers will not receive any Normal IP packets. For these situations we propose that the Ingress/Egress Border Routers should be able to generate dummy IP packets, i.e., IP packets without a payload. These dummy IP packets will be then used as PP, RP, OP or MP packets.

Example: Suppose that the ingress router wants to reserve 10 units of resources during the refreshment period (*i+1*). Then this ingress router will have to send 10 PP packets during period (*i*). The egress router will check for any marked packet. If any MP is detected then the egress edge router will copy the marker from the received packet's header to the header/payload of a reverse packet and sends it back to the initiating ingress edge router. The ingress router at the end of the refreshment period will know how many units of resources were reserved and can be used during the period (*i+1*). Consider now that the ingress router did not receive any MP packets. This will mean that the Diffserv domain reserved 10 units of resources which will be used during period (*i+1*).

Suppose that during period (i+2) the ingress router from this 10 unit of resources would want to use only the 5 unit of resources. Then during period (i+1) the ingress router will send only 5 RP messages. The egress router will check for any marked packet. If any MP is detected then the egress edge router will copy the marker from the IP packet's header to the header/payload of a reverse packet and sends it back to the initiating ingress edge router. Then the ingress router at the end of the refreshment period will know how many units of resources were reserved and can be used during the period (*i+2*).

2.1.6 RSVP Operation within IP tunnels

RSVP operation within IP tunnels, is a mechanism for reserving resources in IP tunnels, in order to extend RSVP usage to fixed and wireless networks [RFC2746]. It enhances IP tunnels with RSVP capability, such that the end-points of the tunnel send RSVP PATH/RESV messages respectively in order to reserve resources for end-to-end sessions within the tunnel. Also the end-points of the tunnel have to map the per-flow end-to-end sessions to the tunnel session.

2.1.7 IETF Intserv/Diffserv Framework

In order to provide scalable end-to-end QoS, RSVP/Intserv and Diffserv architectures can be used as complementary technologies in the access and the core networks respectively. In IETF (see [BeYa00]) an Intserv/Diffserv framework has been specified

The main functionality for the Intserv over Diffserv operation will be performed at the edge devices either at Intserv or Diffserv, i.e. Edge Routers (ER1, ER2) and Border Routers (BR1, BR2), depending on the specific realisations of the framework. These devices will have the burden handling both RSVP/Intserv messages and Diffserv packets.

The specific realisation of the framework depend on whether the resources within the Diffserv are statically or dynamically provisioned: The reference network for the RSVP/Intserv over Diffserv proposed framework [BeYa00] is shown in Figure 2-8.

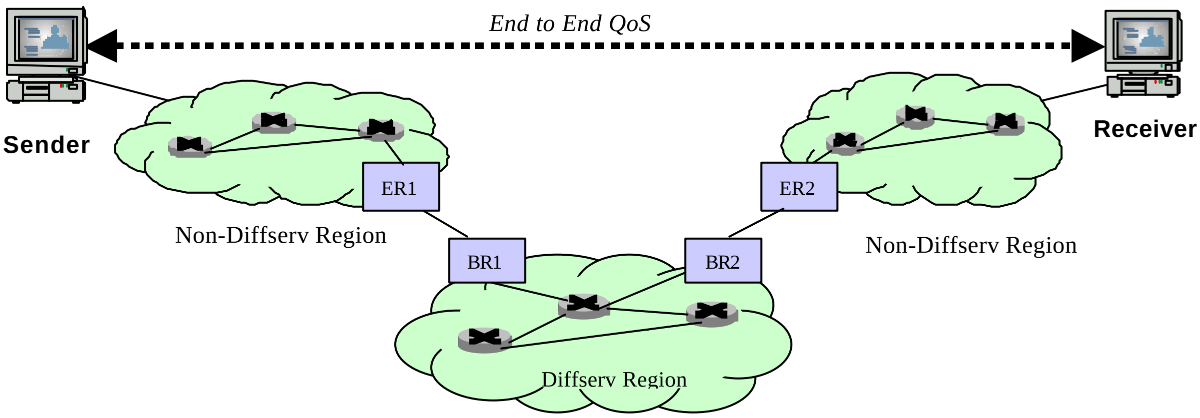


Figure 2-8: Intserv/Diffserv framework

2.1.8 Overprovisioning

One way to overcome the limitations of the best effort model networks in providing QoS, without applying neither Intserv or Diffserv architectures is by overprovisioning the networks, that is by allocating more bandwidth than the expected network peak requirements. Still, even though overprovisioning of network increases the probability of having enough resources available for real-time applications, it doesn't guarantee the quality of service for these applications and on the other hand there will always be a waste of resources in an overprovisioned network.

2.1.9 MPLS

The MultiProtocol Label Switching (MPLS) [RoVi00] is specified by the IETF to mainly be used in combination with the Diffserv concept and is an advanced forwarding scheme that extends routing with respect to packet forwarding and path controlling (see also [XiHa00], [Will99]).

The packet forwarding is used to create topology driven paths through the network. Each MPLS packet has a header that among others can contain a label. An MPLS-capable router, called also Label Switching Router (LSR), accomplishes the forwarding of the packets that it receives by examining the label and possibly another field in the header, called experimental field. At the ingress of an MPLS-capable domain the IP packets are classified and routed based on a combination of the local routing information maintained by the LSRs and of the information carried on the IP header. At this point an MPLS header is inserted for each packet. Each LSR within the MPLS-capable domain will use the label as the index to look up the forwarding table of the LSR. By using the packet forwarding procedure Label Switch Paths (LSPs) are established between pairs of nodes in the network.

The path controlling can be achieved by using traffic engineering. In this case the LSP establishment, maintenance and release is strictly controlled and managed by using signalling. This signalling carries information related to the required characteristics for each LSP. Each node must ensure that these requirements are satisfied. An LSP can include the following requirement characteristics:

- Path Selection (based on QoS constraints);
- Delay and delay variation;
- Bandwidth, including sustained peak data rates and maximum burst sizes;

2.2 Session Protocols

In our framework besides for negotiation, managing and controlling of sessions, we have foreseen the usage of session protocols also as means of session layer negotiation of QoS parameters. Different QoS parameters can be negotiated via the session protocols, e.g., the bandwidth, peak rate, etc. The following section gives a short description of the Session Initiation Protocol (SIP) as the most appropriate for usage in the framework we propose.

2.2.1 SIP

The Session Initiation protocol (SIP) [RFC2543] is an application layer control (signaling) protocol for creating, modifying, and terminating sessions between multiple participants. SIP uses among others two gateway proxies. One of them is called SIP proxy server and the other one is called SIP redirect server. The used messages used in the SIP protocol are listed in Table 2-2.

SIP Messages	
SIP Message Name	**SIP Message Function**
INVITE	Invitation of a user to a session. The message body contains the session description, e.g. using SDP. SDP contains: session name and purpose, time(s) the session is active, the media comprising the session, information to receive those media (addresses, ports, formats and so on) and it may contain additional information about the bandwidth to be used by the conference [RFC2327].
ACK	A confirmation to any response. It is also the only message that doesn't trigger a response upon reception.
BYE	A user is leaving the session.
OPTIONS	Queries a server about its capabilities
CANCEL	Cancel a pending request

REGISTER	Register with a SIP server

Table 2-2: The SIP messages

The SIP proxy server is used to locate the destination party of an invitation to a session and it remains in the signalling path for the duration of the session setup. Figure 2-9 views the SIP session setup when the SIP proxy server is used. Upon the receipt of an INVITE request message the SIP proxy server will query its location database to find out the location of the destination (called) terminal. All the subsequent signalling is routed via the SIP proxy server. The user data is transported through the transport network directly between the sending (calling) and receiving (called) terminals.

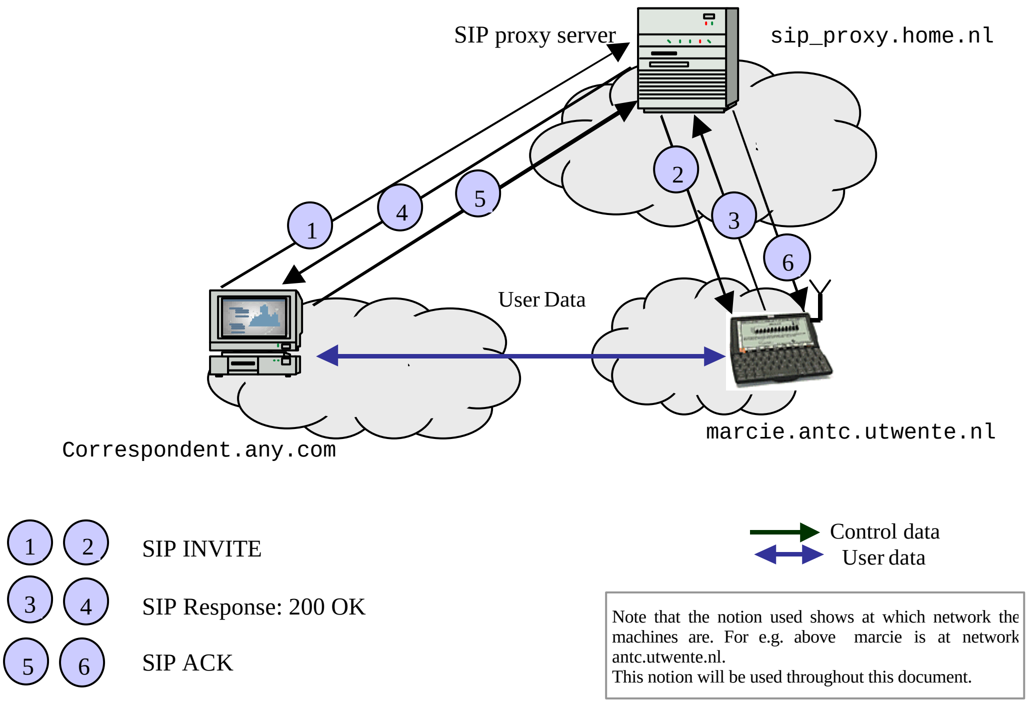


Figure 2-9: SIP session setup using the SIP proxy server

Figure 2-10 depicts the SIP session release procedure when the SIP proxy server is used.

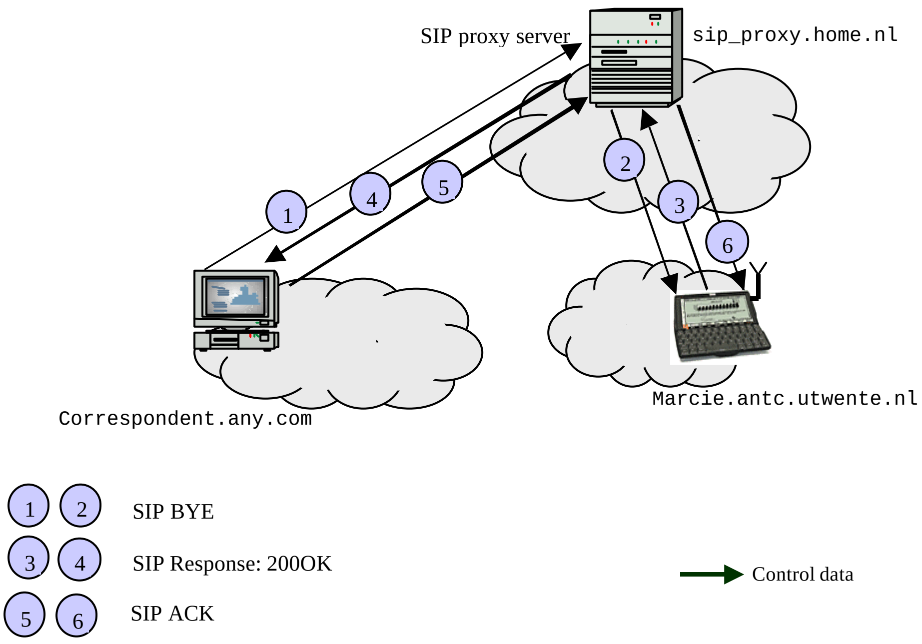


Figure 2-10: SIP session release using the SIP proxy server

Another proxy server used by the SIP protocol is the SIP redirect server. This server (see Figure 2-11) is used to locate the called party to a session but it does not remain in the signalling path for the duration of the session setup. After finding out the location of the called party it sends this information back to the calling party. Afterwards, the user data is transported through the transport network directly between the sending (calling) and receiving (called) terminals.

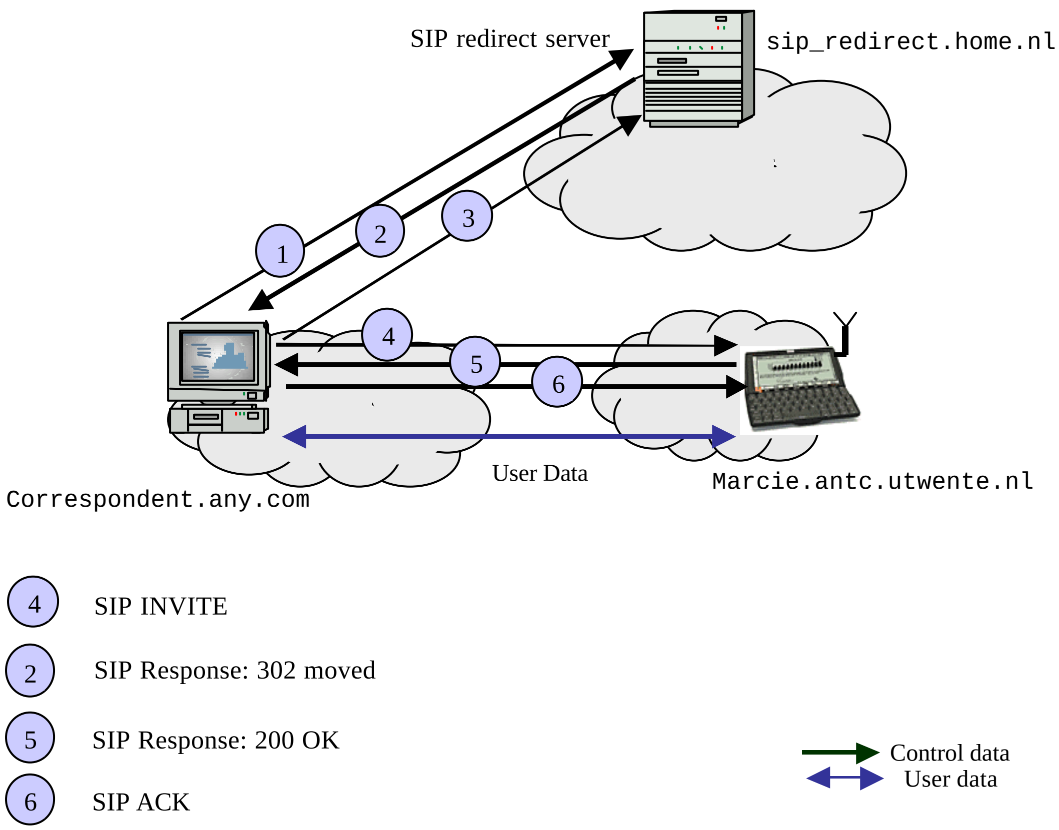


Figure 2-11: SIP session setup using the SIP redirect server

Figure 2-12 depicts the SIP session release procedure when the SIP redirect server is used.

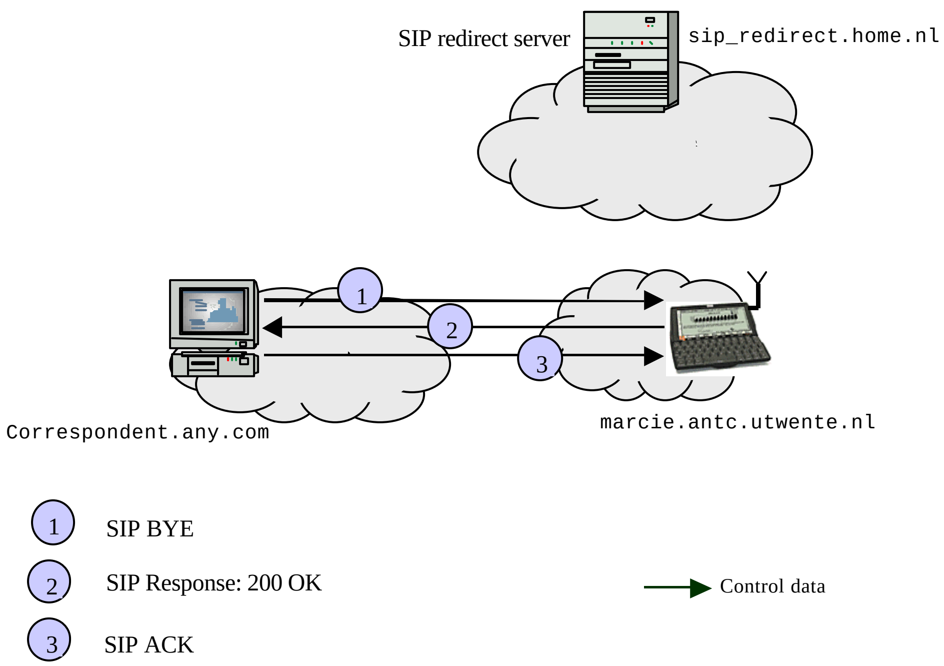


Figure 2-12: SIP session release using the SIP redirect server

2.3 IP Mobility Support

Enabling mobile devices seamless communication and access to the Internet via their wireless network interfaces, independent of their roaming in other networks, requires efficient protocols that will be able to inform the network about the changes in their network attachments.

2.3.1 Mobile IP

The Mobile IP protocol is the most common protocol for providing mobility support at the IP layer, transparently to the layers on top, e.g. TCP. The key feature of the Mobile IP [RFC2002] design is that all required functionality for processing and managing mobility information are embedded in well-defined entities, the Home Agent (HA), Foreign Agent (FA), and Mobile Node (MN). The Mobile IP protocol allows the MNs to retain their IP address regardless of their point of attachment to the network. This can be fulfilled by allowing the MN to use two IP addresses, the home address which is static and is mainly used to identify higher layer connections, e.g. TCP, and the Care-of Address, which has to identify the mobile's new point of attachment with respect to the network topology. In Mobile IPv4 the Foreign Agent manages the Care-of Address. Mobile IP functionality is realized by using three mechanisms (for detailed descriptions of these mechanisms see [Per98] and [Per97]):

- Discovering the Care-of Address (see Figure 2-13): The Care-of address discovery procedure used in Mobile IP is based on the ICMP (Internet Control Message Protocol) Router Advertisement standard protocol, specified in RFC 1256 [RFC1256]. In Mobile IPv4, the router advertisements are extended to also contain the required Care-of Address. These extended router advertisements are known as agent advertisements. Home Agents and Foreign Agents typically broadcast at regular intervals (e.g., once a second, or once every few seconds) and in a random fashion, agent advertisements.

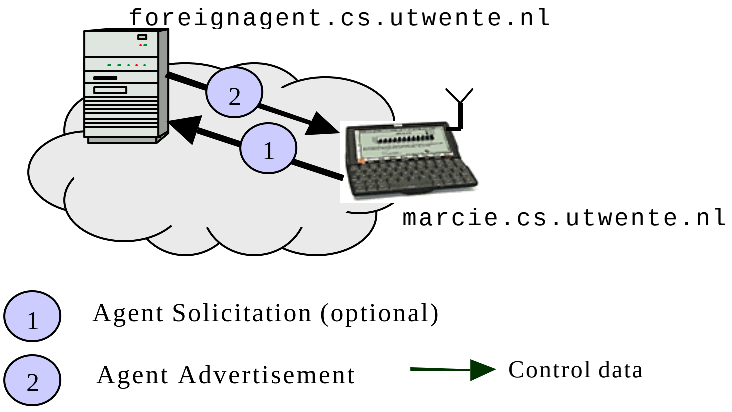


Figure 2-13: Care of Address Discovery

- Registering the Care-of Address (see Figure 2-14): After the Mobile Node gets the Care-of Address it will have to inform the Home Agent about it. In Mobile IP this can be accomplished by using the registration procedure. The Mobile Node sends a registration request (using the User Datagram Protocol (UDP)) with the Care-of Address information. This information is received by the Home Agent and normally, if the request is approved it adds the necessary information to its routing table and sends a registration reply back to the Mobile Node.

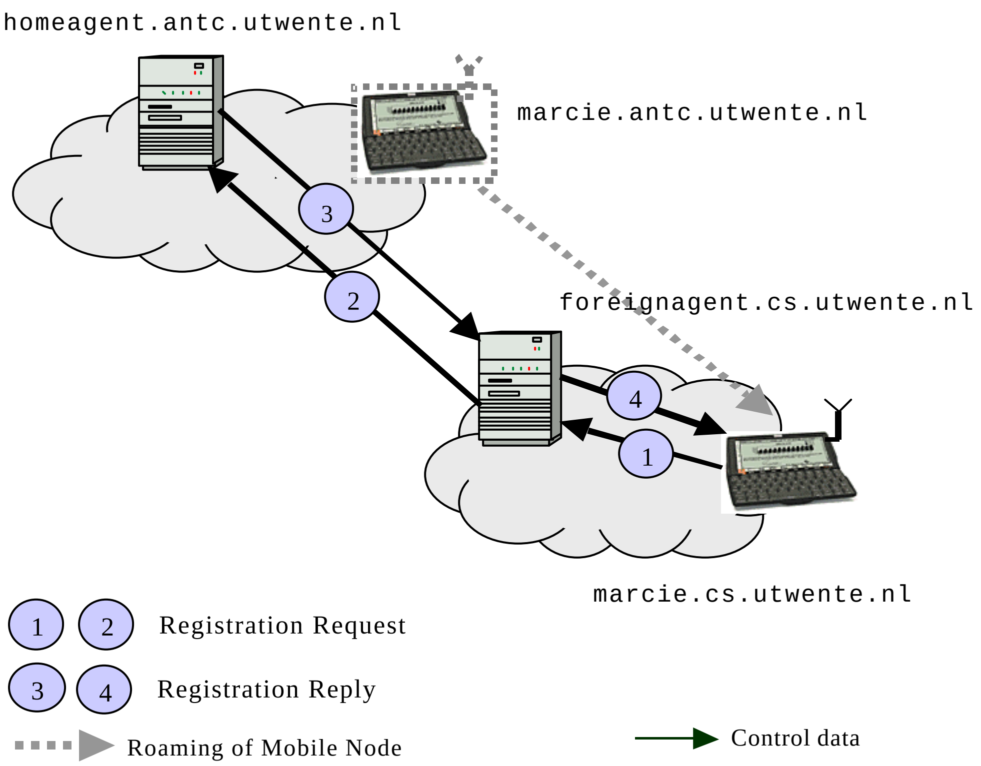


Figure 2-14: Registering the Care-of Address

- Tunnelling to the Care-of Address (see Figure 2-15): is accomplished by using encapsulation mechanisms. All mobility agents, i.e., Home Agents and Foreign Agents, using Mobile IPv4 must be able to use a default encapsulation mechanism included in the IP within IP protocol [RFC2003]. By using this protocol, the source of the tunnel, i.e., Home Agent, inserts an IP tunnel header, in front of the header of any original IP packet addressed to the Mobile Node's home address. The destination of this tunnel is the Mobile Node's Care-of Address. In IP within IP [RFC2003] there is a way to indicate that the next protocol header is again an IP header. This is accomplished by indicating, in the tunnel header, that the higher-level protocol number is '4'. The entire original IP header is preserved as the first part of the payload of the packet. By eliminating the tunnel header the original packet can be recovered.

  When the Mobile IP packet flow, follows a route similar to the one viewed in Figure 2-15, then the routing situation is typically called triangle routing. The packets sent by the host, called correspondent host (CH), follow the path 1,2 and 3, while the packets sent by the Mobile Node follow routes 4 and 5.

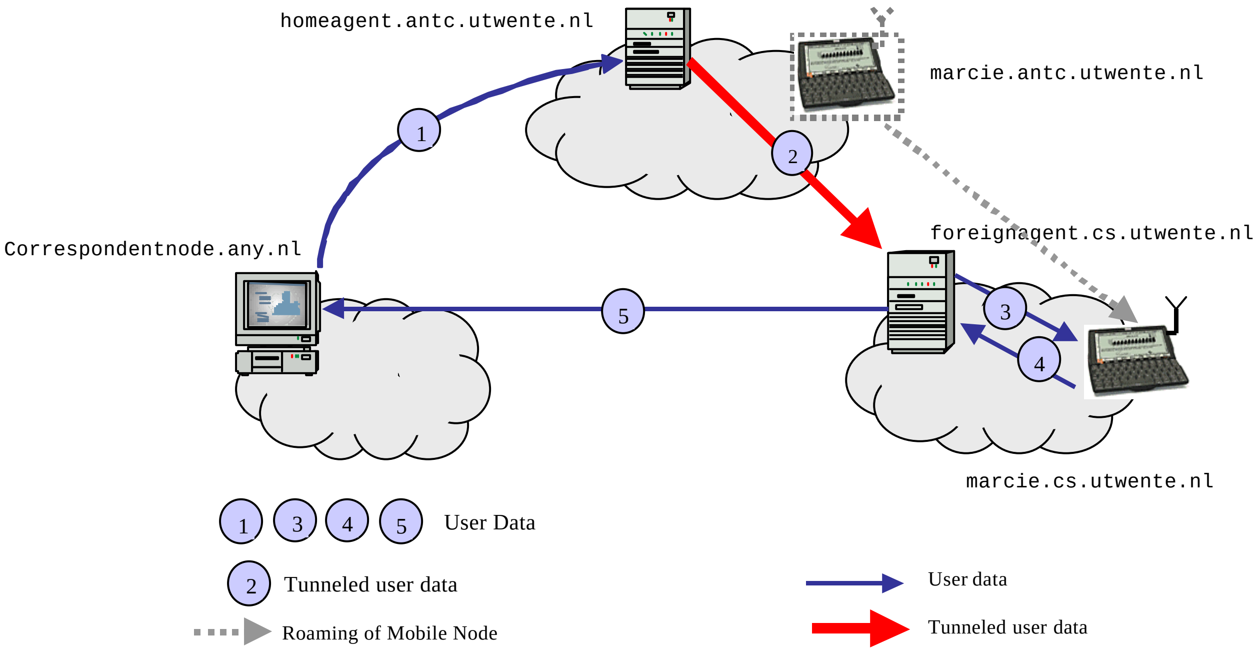


Figure 2-15: Tunneling to the Care-of Address

2.3.1.1 Triangle routing and route optimisation

In [PeJo00] (see e.g., [Per97], [Per98] and [Kar99]), the operation of the base Mobile IP protocol is extended to allow for more efficient routing procedures, such that IP packets can be routed from a correspondent host to a Mobile Node without going to the Home Agent first.

These extensions are referred to as route optimisation (see Figure 2-16), wherein new methods for IP nodes, e.g., correspondent hosts, are provided. The correspondent host receives a binding update message from the mobile's node Home Agent that contains the Mobile Node's Care-of Address. The binding specifies the association of the home address of a Mobile Node with a care-of address for that Mobile Node, along with the remaining lifetime of that association. This binding is then stored by the correspondent host in a binding cache and is used to tunnel its own IP packets directly to the care-of address, bypassing the Mobile Node's Home Agent. In this way, the triangular routing situation, explained in Section 2.3.1 is eliminated. However, in the initiation phase, the IP packets sent by the correspondent host still use the triangle routing until the moment that the binding update message sent by the Mobile Node's Home Agent, is received by the correspondent host.

In addition to the Binding Update message, the route optimisation procedure is using the following messages:

- A *binding warning* control message is usually sent by a node (e.g., Mobile Node or Correspondent Host), to the Home Agent (i.e., recipient), indicating that a Correspondent Host (i.e., target) seems to be unaware of the Mobile Node's new Care-of Address;

- A *binding request* message is send by a Correspondent Host to the Home Agent at the moment it determines that its binding should be initiated or refreshed. Note that if the home agent for a certain reason (e.g. the Mobile Node is in its home domain), can not find or does not want to inform the correspondent host about the MN's Care_of Address, then the Home Agent will send a Binding Update message also to the CH. However, this message will include a Care_of Address that is set equal to the MN's home address and the association lifetime is set to zero. The CH will then have to delete the binding cache entry for that particular MN.

- A *binding acknowledgement* message can be requested by a Mobile Node from a Correspondent Host that has had received the *binding update* message.

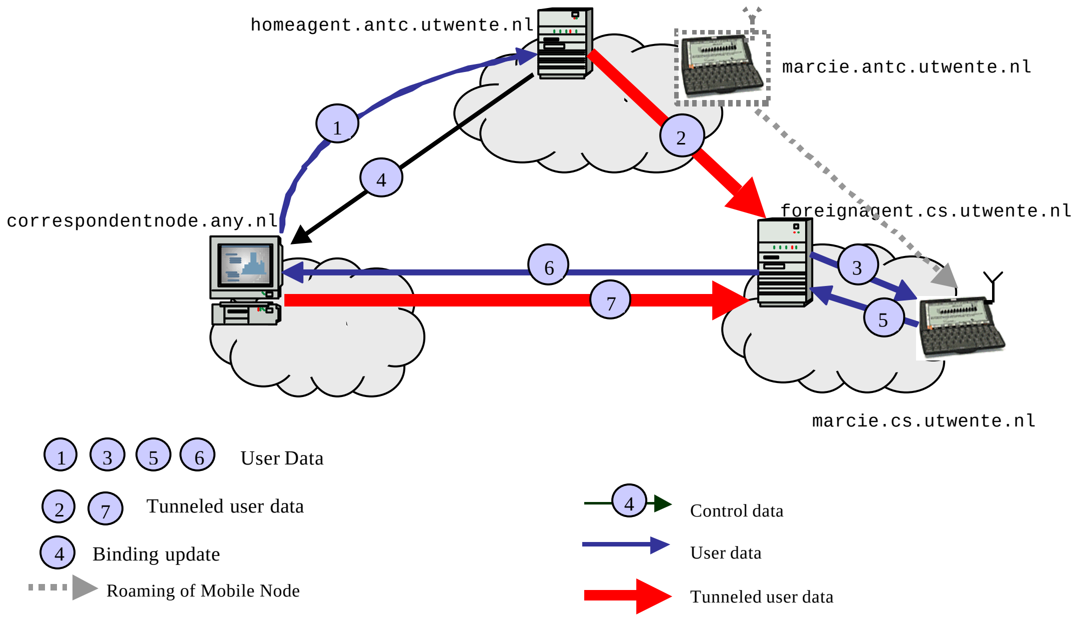


Figure 2-16: Route optimization in Mobile IP

2.3.2 SIP and mobility support

Unlike Mobile IP, the mobility support using Session Initiation Protocol (SIP) [WeSc99] proposes a mechanism in handling mobility at the higher layer, that is the application (i.e. session) layer whenever that is applicable.

Similar to Mobile IP, the mobile host has a home network that is managed by a physical entity called SIP redirect server. The SIP redirect server as explained in Section 2.2, similar to the HA in Mobile IP, is capable of storing information regarding the location of a mobile host. Every time that a mobile host roams into a new IP sub-network it will inform the SIP redirect server about its new IP address that it will register (see Figure 2-17). When a correspondent host wishes to communicate with a Mobile Host, it will send an invite message to the SIP redirect server. The SIP redirect server will send the IP address of the Mobile Host to the Correspondent host. If the mobile host is moving during the session, it sends an invite message to the correspondent host to inform it about its new IP address. The correspondent host will use this IP address to send all the subsequent IP user data traffic to the Mobile Host.

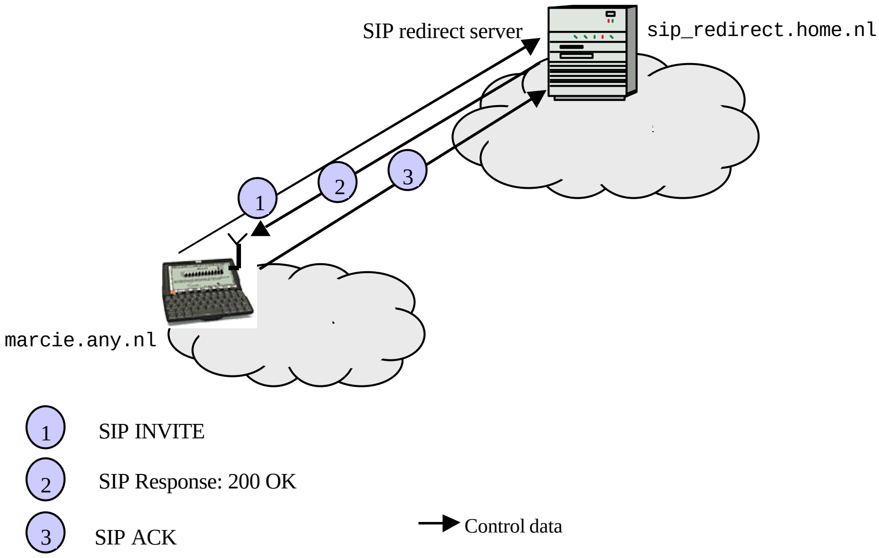


Figure 2-17: IP address registration at the SIP redirect server

The operation of SIP mobility support is shown in Figure 2-18 and the explanation of the SIP messages is given in Table 2-2. When the mobile host (e.g.marcie.home.nl) is moving from its home network to another location (e.g. marcie.foreign1.nl or marcie.foreign2.nl) than it always registers its new location with the SIP redirect server, similar to home agent registration in Mobile IP (see Figure 2-18 msg. 9,10). When the correspondent host (e.g. correspondent.any.nl) which might be another mobile host or an fixed host begins a session, it sends an INVITE to the redirect server which will then redirect this message to current location of the mobile node (see Figure 2-18 msg. 1-5). When the mobile host is moving during the ongoing session than it must send a new INVITE to the correspondent node (see Figure 2-18 msg. 6-8). In the Contact field of the SIP message it inserts a new IP address, which the correspondent node will indicate in the new SDP field as a transport address for redirecting the data flow.

The advantages of mobility support using SIP instead of Mobile IP are: there will be no need for tunnelling data packets, easily applicable to most common applications (that are not using the TCP protocol) and thus no need for changes of the IP protocol stack of the mobile host. However, the SIP mobility cannot support TCP connections, which limits its usage only for real-time communications using UDP.

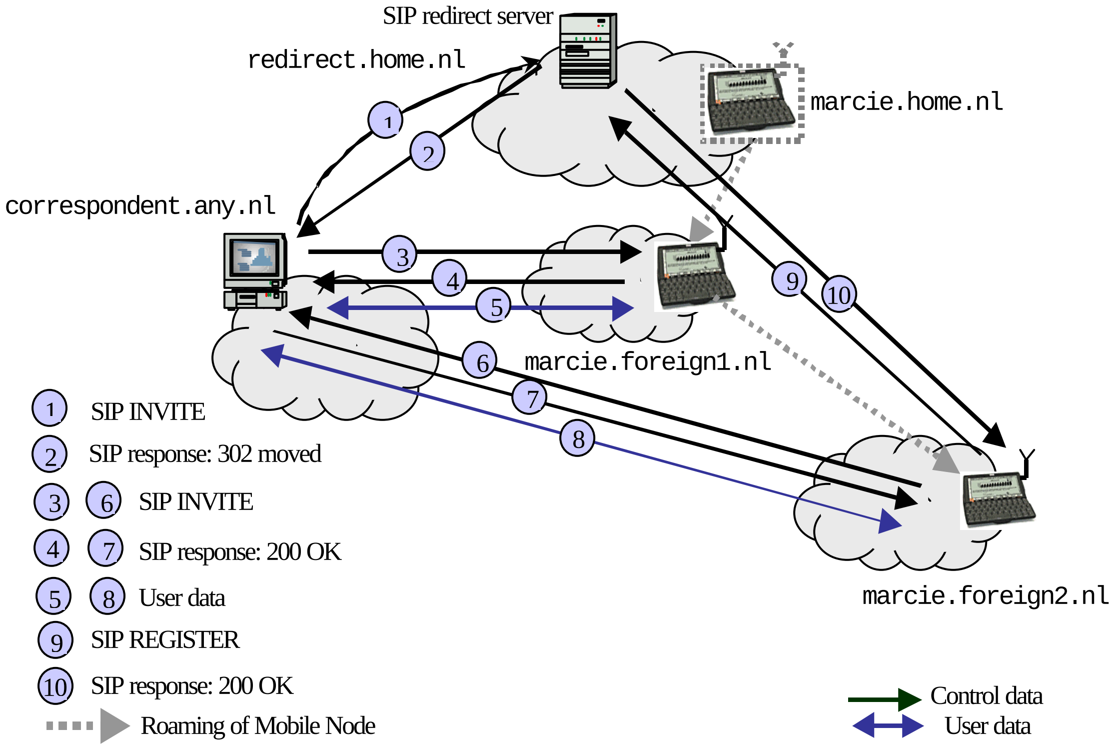


Figure 2-18: SIP mobility support

3 General Description of QoS and Mobility Framework

The QoS and mobility framework architecture that we introduce is initially intended to be a flexible and open architecture suitable to be applied for a large variety of applications with different QoS demands, different wireless and wired access technologies and different protocols. In this section we introduce the requirements that this framework will have to support and present the functional entities and protocols used. We also identify QoS Service Mobility classes, to make a relation between the known QoS service classes and the mobility of host. Further we give an overview of UMTS (Universal Mobile Telecommunications System) QoS classes and propose their mapping to QoS mobility classes defined.

3.1 Design Goal and Requirements

The next generation Internet will have to support a large variety of applications with different QoS demands that are running on different types of wireless or wired terminals connected on various types of networks. This requires that the next generation Internet architecture will have to be very flexible and open, capable of supporting all these different types of networks, terminals and applications.

Furthermore, it can be seen that existing QoS management architectures are optimised to operate efficiently in small access networks. Some examples that we found in the literature are as follows:

- Heidelberg *QoS* model [VoWo96] developed by the HeiProject at IBM's European Center in Heidelberg. This model provides guarantees in the end systems and core network;
- Extended Integrated Reference Model (XRM) [LaBh94] developed by the COMET group at Columbia University, that is a modelling framework for control and management of multimedia telecommunications network;
- OMEGA [NaSm95] developed by the University of Pennsylvania. It is examining the relationship between application *QoS* requirements and the ability of local and resource management to satisfy these demands;
- Quality of Service Architecture (*QoS*-A) [CaCo93]: describes a layered architecture of services and mechanisms for quality of service management and control of continuous media flows in multi-service networks.
- OSI *QoS* Framework [ISO95] concentrates primarily on *QoS* support for OSI communications. This framework provides a model, which identifies objects of interest to *QoS* in open system standards.
- TINA *QoS* Framework describes a framework for specifying *QoS* Aspects of distributed telecommunications within the context of the Computing Architecture.

- MASI end to end model [BeDa94] developed by the CESAME project at laboratories MASI at Universite Pierre et Marie Curie, is developing an architecture for multimedia communications which takes end to end *QoS* support as it primary objective.

It is therefore reasonable to consider that a framework should provide the opportunity to support efficient local access QoS management architectures. Furthermore, regarding the QoS solutions provided in the core networks we believe that a flexible and scalable architecture should be used. In several papers and reports, (e.g. [RFC2475], [RFC2638]) it is claimed that the Differentiated Services architecture is a flexible and scalable QoS architecture that should be used in the core network of the next generation Internet . We think that their claim is valid.

Based on the above given considerations we have created a list of requirements that should be fulfilled by our proposed framework architecture:

- The IP core network is based on either the Diffserv network architecture or a mix of Diffserv and overprovisioned IP core networks. The second option is only valid if the provider of IP overprovisioned networks guarantees certain QoS bounds.

- Both static and dynamic provisioning of resources in the IP Diffserv core network should be supported.

- The access networks may support any of the existing IP QoS management architectures, like Integrated Services Architecture, Differentiated Services Architecture, QoS capabilities of the access technology, overprovisioning of resources, etc. In the situation that an access network operator configures its network in such a way that it becomes overprovisioned, applications may or may not gain the demanded QoS.

- The access networks may support different access technologies, e.g. Bluetooth, General Packet Radio Service (GPRS), Universal Mobile Telecommunications System (UMTS), Wireless Local Area Network (W-LAN).

- Each mobile node that supports multiple access technologies should be able to select the most efficient and cost-effective technology that supports the application QoS requirements.

- Handovers between different access networks and technologies should be supported.

- Global QoS interoperation of local QoS mechanisms should be possible.

3.2 Separation of QoS Session Negotiation and Resource Reservation

Especially in a wireless and mobile environment, it is very important to be able to separate the negotiation of a communication session and its QoS from the actual reservation of the communication resources. In wireless networks the available resources are scarce, and therefore, efficient resource reservation mechanisms should be applied.

Efficient resource reservation mechanisms should reserve resources only when it is certain that these resources will be used. Furthermore, the scalability of a network and in particular the scalability of a large IP core network will be enhanced if its resources are only reserved when it is certain that they will be used.

A separation between session layer control (or negotiation) and bearer control (or resource reservation) as proposed in [ErOh00] is justified, since negotiating service at the session level certainly adds value to the QoS and mobility framework for several reasons:

- Session negotiation can establish the session before claiming the resources. This avoids an unnecessary reservation of communication resources due to unavailability of a suitable (e.g. high speed, when a mobile host is "on the move") access network, incompatible session / application layer parameters, shortage of resources in the remote access network, or a remote user not accepting the invitation.

- Separation of session negotiation and resource reservation allows for mobility issues to be sorted out before resources are reserved. For instance, the (Care-of) IP address of a mobile host can be obtained before any resources are reserved.

- In the Internet protocol suite, separate protocols are available for session control and resource reservation. These are e.g., SIP for session control and RSVP or Load Control for resource reservation.

3.3 QoS Mobility Service Classes

Both architectures Intserv and Diffserv define service classes that can be used by different types of applications. Applications that require hard QoS guarantees for their operation, such as intolerant real time applications [RFC1633], e.g. Voice over IP (VoIP), we will call non-adaptive applications. For these applications the Intserv architecture recommends the Guaranteed service model, while Diffserv architecture defines the EF-PHB [RFC2598] to support them. Certain real-time application, which are tolerant [RFC1633], e.g. one way voice or video, will require for their operation soft QoS guarantees, i.e. they may be tolerant in terms of delay bounds and jitter. We call such applications adaptive. For these applications the Intserv architecture recommends the Controlled Load service model, while Diffserv architecture defines the AF-PHB [RFC2597] to support them.

These service classes do not include the support for IP mobility and therefore roaming users will not be able to use applications with a satisfactory QoS. Therefore, in this paper we propose and specify two new QoS service classes, which are extensions of the aforementioned classes:

- *Mobility Dependent Locally Guaranteed (MDLG)* is associated with non-adaptive applications. In this service class the QoS requirements can be statistically guaranteed locally in a subnetwork[*]. The statistically guarantees of a QoS requirement are related to a probabilistic guarantee, e.g., a QoS requirement can be guaranteed with a certain probability, e.g., 95%.

[*] The definition of the subnetwork in this case is the same as defined in Mobile IP [Per98].

When the mobile host moves to another IP subnetwork that provides a lower QoS, the application will re-negotiate the QoS parameters by specifying the lowest QoS limit that the application is willing to accept. Note that when a mobile host moves to another IP subnetwork then the handover requests succeed with higher probability than the new user requests. If the negotiated QoS is lower than its limit than the application terminates the session.

- *Mobility Dependent Adaptive (MDA*) is associated with adaptive applications. This class consists of several relative sub-classes assigned to different QoS levels subsequently. The MDA is similar to the AF-PHB [RFC2597] defined by the Diffserv architecture and to the Controlled Load [RFC1633] defined by the Intserv architecture. When the mobile host moves to another IP sub-network and if the sub-network satisfies the QoS requirements then the application continues with the same QoS, otherwise it adapts to another sub-class with lower QoS. Afterwards, all the other hosts that are probably connected to the roaming host have to be informed about the reduction in the QoS. Furthermore, the handovers succeed with higher probability than the new user requests of the same sub-class. If there are no resources available for none of its sub-classes than its traffic is treated as best effort traffic.

- *Best Effort* is associated with applications requiring no QoS like file transfers or e-mail. No special provisions are taken for moving hosts.

3.4 UMTS QoS Classes

UMTS will become a very significant third generation mobile communication technology. Therefore, we will briefly describe the UMTS QoS classes and will also briefly mention how these QoS classes can be mapped to the proposed QoS mobility classes.

The UMTS QoS classes (see [3GPP 23.107]) are defined depending on delay sensitivity of the user data traffic used by certain applications. These are:

- Conversational class: this class represents conversational streaming applications, e.g., telephony speech that is very delay sensitive. Furthermore, they preserve time relation between the information entities of the stream;

- Streaming class: this class represents real time streaming applications that are not conversational. This could for example be the situation that the UMTS user wants to listen to real time speech or real time video.

- Interactive class: it represents all the non real time applications, such as Web browsing, server access. The main characteristics are the use of request reply pattern and preserve payload content.

- Background class: This class represents all the applications that are the most delay insensitive. In other words the data that is sent by such an application can be processed in the background. Such applications are e.g., e-mail, SMS (Short Message Service).

3.4.1 Mapping of the UMTS QoS classes to Mobility Service classes

The UMTS QoS classes are actually specified at the application layer. These QoS classes have to be mapped either directly to bearer QoS classes or to network QoS classes. In this section we only concentrate on the situation that the UMTS QoS classes are mapped to network QoS classes and in particular to the QoS Mobility Service classes defined in Section 3.3.

This mapping is listed in Table 3-1. We consider that this mapping can be considered as straightforward.

The real time conversational and streaming services require very low delay and delay variations in their user data traffic. The MDLG and MDA QoS Mobility Service classes could satisfy both requirements. Of course MDLG can better support these UMTS QoS classes, since it is capable of providing statistically guarantees. If the user wishes to receive a service with a guaranteed QoS then the MDLG will be used in the QoS mapping, otherwise MDA could be used.

The interactive QoS classes are demanding less QoS requirements in delay and delay variation of the user data traffic. Therefore, we propose that either MDA or Best effort services can be applied to this QoS mapping. The reason of choosing the MDA class is because we are considering the situation that the user may like to introduce QoS differentiation between several interactive service classes. We consider that the background traffic can be mapped to the Best effort class.

UMTS QoS Application class	**QoS Mobility Class**
Real time Conversation	MDLG/MDA
Real time Streaming	MDLG/MDA
Interactive Services	MDA/Best effort
Background traffic	Best effort

Table 3-1: Mapping of the UMTS QoS classes to QoS Mobility Service classes

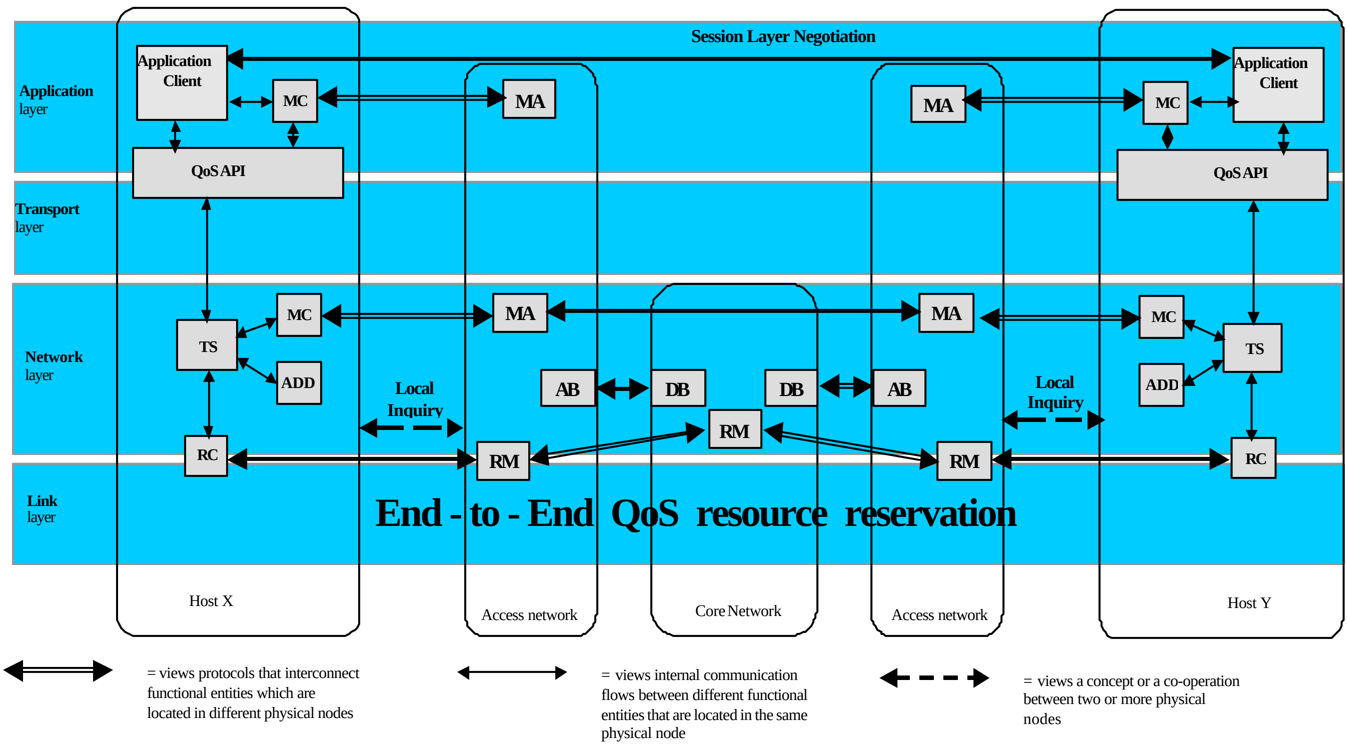
3.5 Framework Entities and Protocols

The QoS and mobility framework depicted in Figure 3-2 consists of three major building blocks: Hosts, local Access Networks, and a Diffserv Core Network. Hosts represents the calling and called hosts, i.e. Host X and Host Y, respectively. The local Access Network includes possible efficient local QoS mechanisms. The Diffserv Core Network is represented as one Diffserv domain, but it may consist of more than one Diffserv domains. Each block includes the main active functional entities that have to be used in the QoS and mobility framework. Furthermore, Figure 3-2 depicts the allocation of these functional entities into an TCP/IP protocol suite. The TCP/IP protocol suite is described in e.g., [Stev97] and is depicted in Figure 3-1.

In the following subsections we will first describe examples of protocols that may be used to provide the communication between the applied functional entities which are located in different physical nodes and second we will describe each functional entity per block.

Application (e.g., e-mail, VoIP)
Transport (e.g., TCP,UDP)
Network (e.g., IP)
Link (e.g., Bluetooth)

Figure 3-1: TCP/IP protocol stack



AB	Access Boundary	RC	Resource Client
ADD	functional entity that could be added in the future	RM	Resource Manager
DB	Diffserv Boundary	TS	Technology Selector
MC	Mobility Client	QoS API	Quality of Service Application Program Interface
MA	Mobility Agent		

Figure 3-2: QoS & Mobility Framework building blocks and protocols

3.5.1 Protocols

The following examples of protocols may be used to interconnect the various functional entities in the QoS and Mobility framework.

Session Layer Negotiation protocol is any protocol that the Application entities will use for initiating a session between hosts. It might be SIP or H.323, or it might be an entirely new protocol, as long as it fits within the framework requirements.

End-to-End QoS Resource Reservation protocol (see Section 2.1) is a protocol that will be used for resource reservation in the end-to-end path. It might be RSVP, Load Control, RSVP aggregation, tunnelled RSVP, or a combination of the former protocols.

The Local Inquiry views the information exchange between the host and the access network. It is mainly used for local resource inquiry (see Figure 3-3), i.e. communicating with the access network resource manager. This information exchange can be implemented using various protocols e.g., resource reservation protocols such as RSVP, network management protocols such as SNMP [RFC1905] or COPS [RFC2748], or mobility management protocols such as Mobile IP.

The mobility management protocol is any protocol that can provide the network mobility management, e.g., location management and handover between different IP sub-networks. This can for example be the Mobile IP or the SIP (mobility) protocols.

3.5.2 Host Functional Entities

The following host functional entities are required for the QoS and mobility framework:

- Application Client is realised by the Application layer and it is an abstraction of a QoS aware application. The QoS aware application is any application that is able to specify its traffic and QoS requirements, based on which the QoS API determines to which QoS Mobility Service Class it belong, i.e. its service profile. It is also required that the application client support the session layer protocols. When e.g., the application is SIP then the application client will be a SIP client.

- QoS API is the abstraction of mechanisms that based on application attributes (e.g. audio, video) and QoS requirements, determine the application's service profile. It will perform mapping of the application service profile in an understandable form for the underlying host Resource Client (RC) and also the mapping of RC messages in an understandable form for the application itself to let it know whether the session initiation is going to be performed or not. Of course these mechanisms should be able to detect when the host has entered another access domain, e.g. using the Mobility Client. (See also [ErOh00] for a similar QoS API definition). This functional entity is realised by the co-operation of the Application and Transport layers. However, the QoS API should be able to send and receive information from the Network and Link layers.

- The host Resource Client (RC) is the abstraction of the entity that is in fact a QoS decision point for the end host. It will provide the mechanism for resource control within the end host based on request and responses it receives from the QoS API, the Technology Selector (TS) and the Local Inquiry protocol messages. The RC should interpret the QoS Mobility Service class parameters and based on their interpretation and the Local Inquiry protocol messages it should decide on whether there are enough network and link resources locally for the Application to initiate a session. This functional entity is realised either entirely by the Network layer or by the co-operation of the Network and Link layers.

- The Mobility Client (MC) is a functional entity that in combination with the Mobility Agent located at the access networks is providing IP mobility management. The Mobility Client is realised either by the Network layer, e.g., Mobile IP, or by the Application layer, e.g., SIP mobility.

- Technology Selector (TS) is the entity, which will be part of any mobile host that wishes to select a certain underlying radio technology and/or underlying wired technology supported by an access network. The TS is able to provide this selection by using certain criteria, based on e.g. application's service profile, mobility scenario, available resources, authentication and accounting scenarios. The TS functional entity is realised by the Network layer.

- ADD (see Figure 3-2) represents any other functional entity that could be added to the framework and that should be located into the Host. Such functional entities could be the authentication and accounting management functional entities.

Figure 3-3 depicts the situation that the host is able and is willing to perform the technology selection. In this situation the host is capable of selecting one of the underlying radio technologies, e.g. Bluetooth and GPRS (General Packet Radio Service). The main operation is as follows.

The Host needs to start a real time application, e.g. VoIP. The QoS API will perform the mapping of the application requests to parameters that are understood by the TS. If the TS entity has the required profile information to perform the technology selection it will do so and it will inform the application entity (i.e. session client) about it. Otherwise, the TS will send one request, i.e. TS_Inquiry REQUEST to the Bluetooth access technology and another request to the other access technology, e.g. GPRS. Note that the TS_Inquiry Request may be sent in either one or more than one messages. These requests will include query information regarding for example: (1) the requested QoS parameters, (2) the authentication restrictions, (3) accounting restrictions, (4) the financial and complexity cost of a connection to the core network, etc. This query information will have to be distributed to all functional entities, e.g. RM, MA, in the access technologies that will be able to answer them. The replies of each queried functional entity will be either sent individually in one TS_Inquiry RESPONSE or they will all be combined in one TS_Inquiry RESPONSE and sent to the Host TS. The TS by applying the predefined criteria will choose one of the access technologies and it will inform the application entity (i.e. session client) about it.

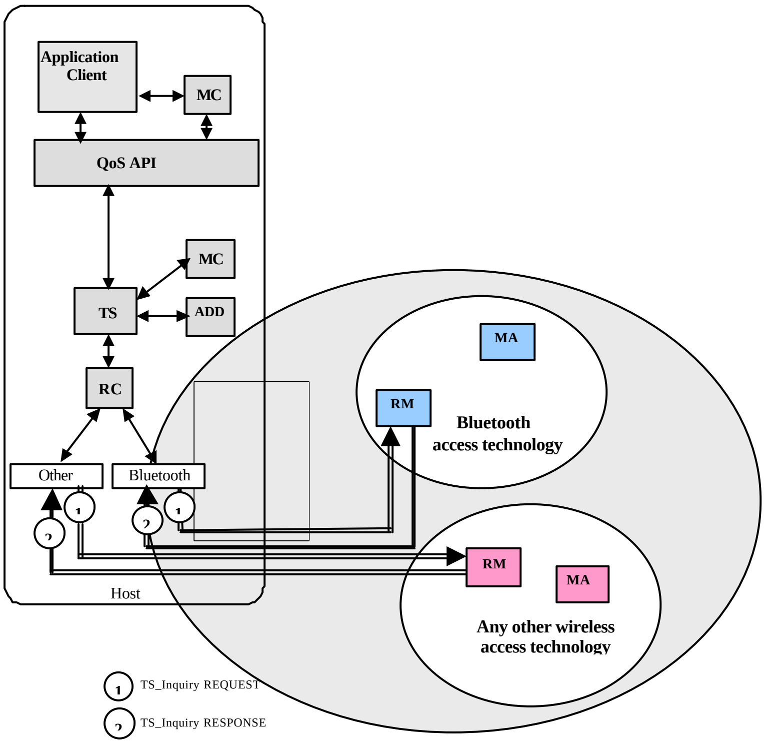


Figure 3-3: Example of technology selection accomplished by the host

3.5.3 Diffserv Core Network Functional Entities

In our requirements we note that the IP core network should be either Diffserv network architecture or a mix of Diffserv and overprovisioned IP core networks. The second option is only valid if the provider of the IP overprovisioned networks, guarantees certain QoS bounds.

Therefore, the functional entities that will be located in this block and are used in the QoS and mobile framework should be in full compliance with the Diffserv network architecture definitions. The functional entities that are located in the Diffserv core network region are:

- The Resource Manager (RM) performs the resource allocation and admission control for the core network either statically or dynamically. We assume that it can be centralised (e.g. Bandwidth Brokers, [RFC2638] or [TeCh99]) or distributed within the core network (e.g. see [RFC2475]). This functional entity is realised by the Network layer.

- The Diffserv Boundary (DB) represents the functionality that is available in standard Diffserv border routers (see e.g., [RFC2475]) and that is managing traffic aggregates from the adjacent domains in compliance with the SLS agreement. In some particular cases it might also perform other tasks for interoperation with other, non-Diffserv domains. This functional entity is realised by the Network layer.

3.5.4 Access Network Functional Entities

The functional entities that are located in a local Access Network and are necessary for the QoS and mobility framework are the following:

- The Resource Manager (RM) functional entity is applied in the access network and similarly to the Diffserv core network Resource Manager is responsible for resource allocation and admission control. Its specific realisation depends on the IP QoS architecture that will be used at the access network and it is realised either entirely by the Network layer or by the co-operation of the Network and Link layers.

- Access Boundary (AB) represents the functionality that is available in any edge device, e.g., Edge Router (ER), residing at the periphery or boundary of an administrative domain. Its functionality depends on specific IP QoS architecture used at the access network and the architecture of the ingress routers of the Diffserv domain. This functional entity is realised by the Network layer.

- Mobility Agent (MA) is an abstraction for all the mechanisms that are related to the IP mobility protocols, e.g. Mobile IP and SIP. It may for example represent a Home Agent or a Foreign Agent or a SIP redirect server (SIPS). This functional entity is realised by the Network layer for the situation that the MA represents a HA or a FA. Moreover, when the MA represents a SIP redirect server then its functionality is realised by the Application layer.

4 Proposed IP QoS and mobility integration

In this section we present the integration of IP QoS architectures, protocols supporting mobility and session protocols, which at this moment we consider relevant for the framework we propose.

4.1 Mobile IP and SIP

In Section 2.3.2 a scenario has been presented wherein the SIP protocol is used to provide mobility support to a mobile node that is roaming between different IP subnetworks. However, the SIP mobility cannot support TCP connections, which limits its usage only for real-time communications using UDP. This disadvantage is solved by combining the SIP and Mobile IP protocols (see Figure 4-1). In this combination SIP is used on top of Mobile IP, in which case Mobile IP provides to SIP the same IP addressing transparency as it provides to TCP. The individual flow diagrams used for the SIP session setup and Mobile IP tunnelling to the Care-of Address are viewed in Figure 2-9 and Figure 2-15, respectively. The messages in Figure 4-1 are used as follows. Messages (1) to (7) describe the SIP session setup procedure. Messages (8) to (12) describe the Mobile IP tunnelling to the Care-of Address procedure. Note that specific access networks to the Internet, such as cellular networks have their own mobility support. However, at the moment, they do not provide mechanisms for roaming between different types of access networks.

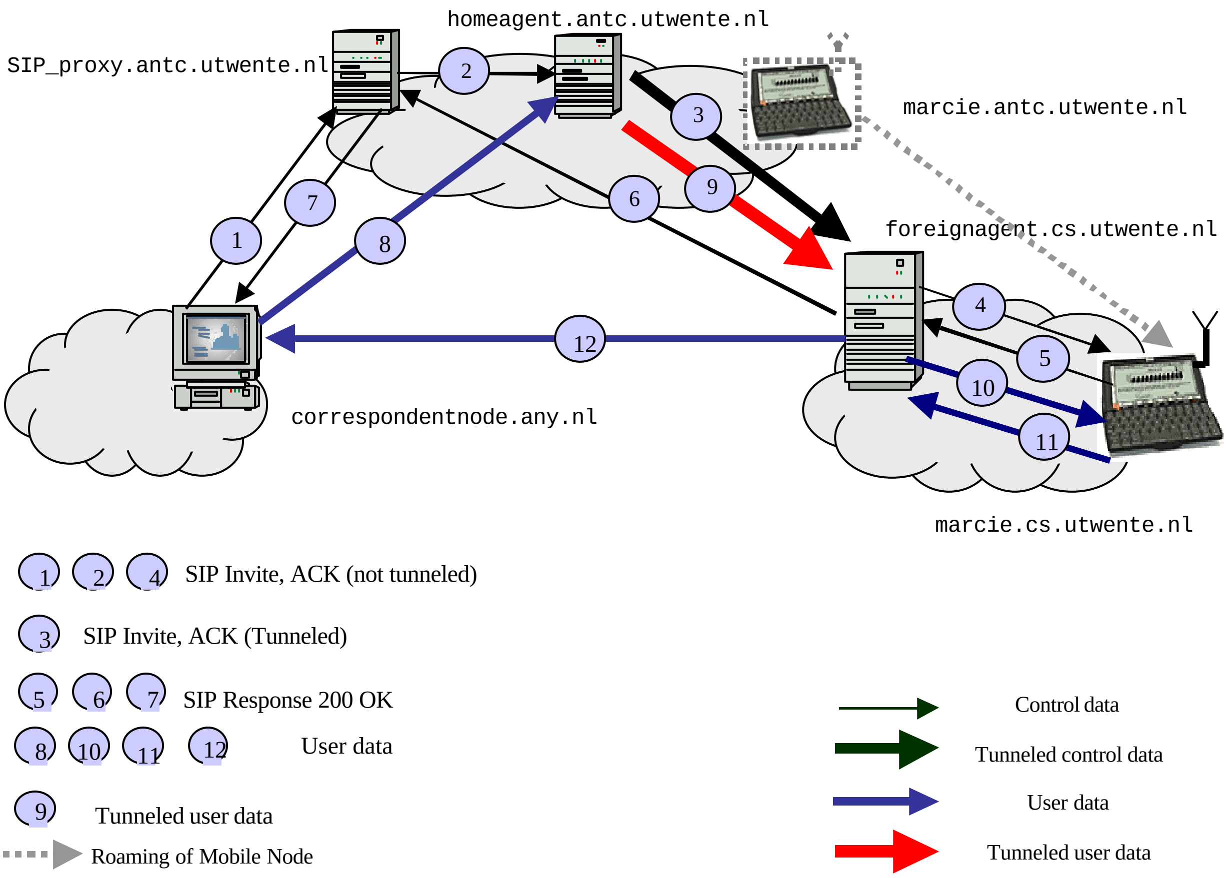


Figure 4-1: SIP session setup and Mobile IP tunneling to the care of address

4.2 RSVP and Mobile IP

It is expected that real time applications running on a mobile node, will need to use the combination of a specific protocol used for IP mobility and another one used for resource reservation, e.g., RSVP. As explained in Section 2.3.1, Mobile IP is the most common protocol used for IP mobility.

By combining the capabilities of these protocols it will be ensured that first, the application will not be terminated when the MN will move into another IP subnetwork and second, the required QoS will be satisfied.

In Figure 4-2 we present the interoperability scenarios between the Mobile IPv4 and RSVP protocols. In this scenario it is assumed that the Mobile Node is roaming into a foreign domain. The Mobile Node has discovered its Care-of Address via the Foreign Agent. Furthermore, its Care-of Address is registered with the Home Agent.

Messages (1) to (6) are used to reserve the resources on the upstream direction, i.e., from Correspondent Host to Mobile Node. For the downstream direction, i.e., from Mobile Node to Correspondent Host, the resource reservation is accomplished using messages (7) and (8). Subsequently, the IP user data transfer on the upstream and downstream directions takes place. At the moment that one of the end terminals is willing to terminate the application, then the resource release procedures are initiated. Messages (9) to (14) are used to release the resources on the upstream direction, while messages (15) and (16) are used to release the resources on the downstream direction.

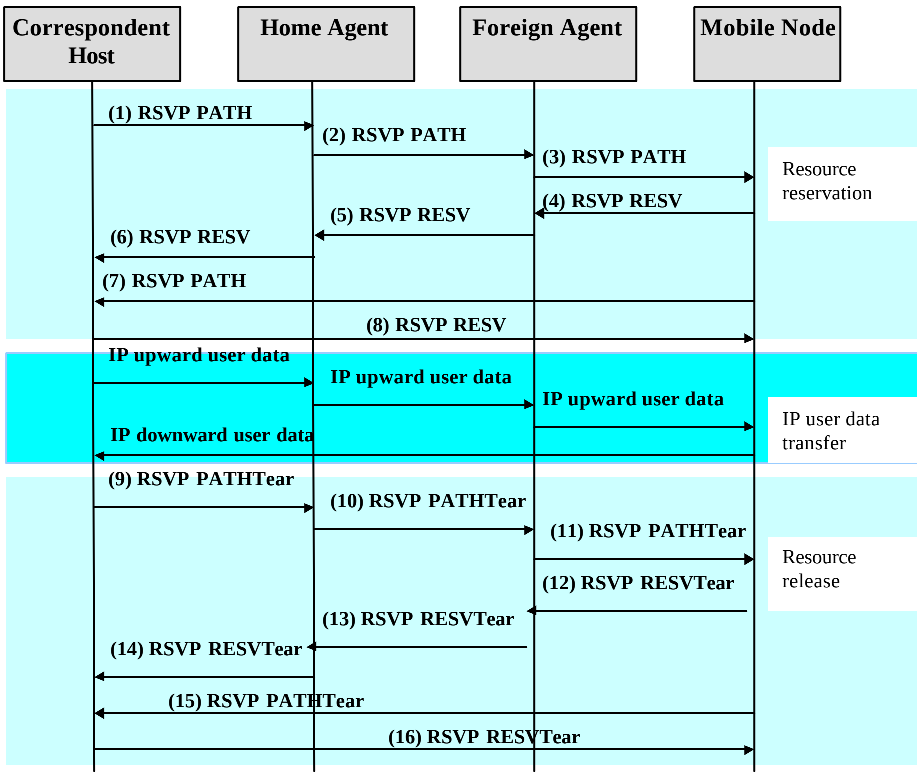


Figure 4-2: Interoperability Mobile IPv4 and RSVP

4.3 RSVP and SIP (used for mobility support)

By combining the capabilities of the RSVP and SIP (used for mobility support) protocols it is ensured that first the required QoS is satisfied and second that the session in progress will not be terminated when the MN will move into another IP subnetwork.

In Figure 4-3 we present an interoperability scenario between the SIP and RSVP protocols. Messages (1) to (6) are used to setup the session between the Correspondent Host and the Mobile Node. The messages (7) and (8) accomplish the resource reservation for the upstream direction. For the downstream direction the messages (9) and (10) accomplish the resource reservation. Subsequently, the IP user data transfer in both directions, upstream and downstream, takes place. When one of the end terminals wishes to terminate the application then a combination of resource release and a session release procedure will be accomplished. Messages (11) to (14) are used to release the resources on both the upstream and downstream directions. Messages (15) to (17) are used for session release.

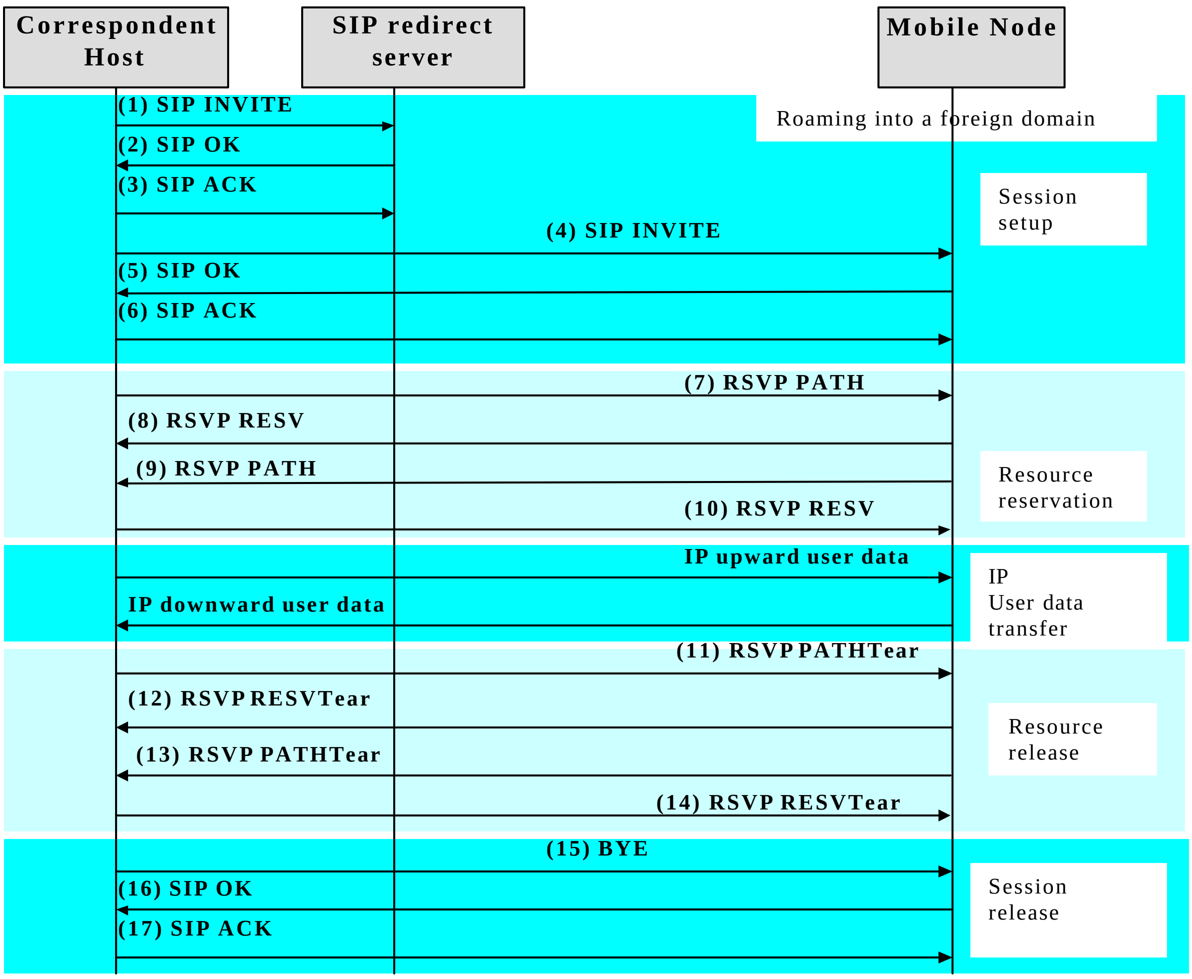


Figure 4-3: Interoperability between SIP (used for mobility support) and RSVP

4.4 Load Control and Mobile IP

In Figure 4-4 an interoperability scenario between the Load Control and Mobile IP protocols is presented. Messages (1) to (3) are used to reserve the resources between the Correspondent Host and the Mobile Node. The message (4) accomplish the resource reservation between the Mobile Node and Correspondent Host. The reservation is achieved by using the Load Control protocol (see Section 2.1.5. Subsequently, the IP user data transfer in both directions, upstream and downstream, takes place. When one of the end terminals wishes to terminate the application then a Load Control resource release will be accomplished. Messages (7) to (12) show that the reserved resources are not refreshed, i.e., no RP messages are sent form an ingress node to an egress node. This will cause the release of the resources that are not refreshed.

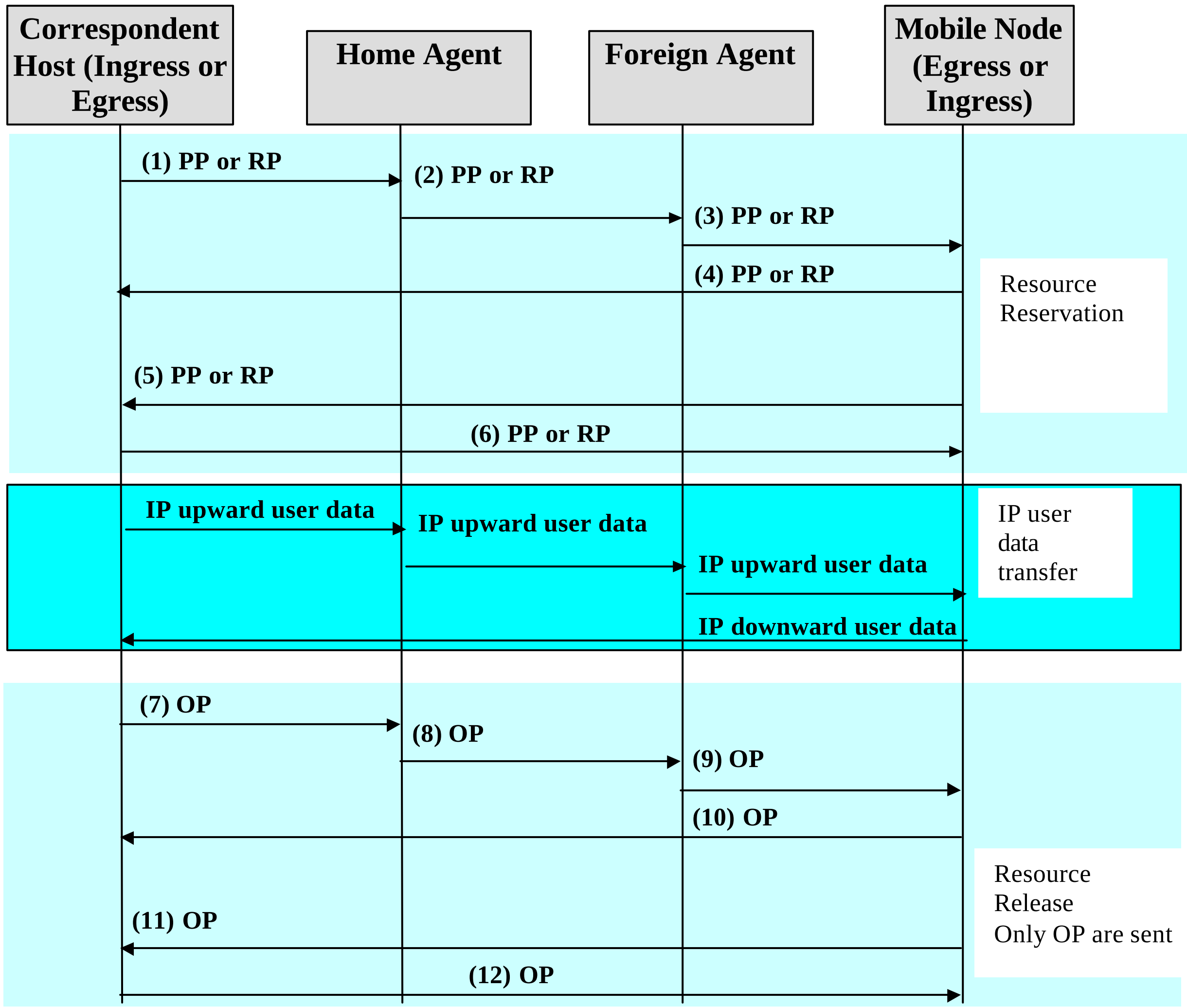


Figure 4-4: Interoperability between Mobile IP and Load Control

4.5 Load Control and SIP (used for mobility support)

The interoperability scenario between the Load Control and the SIP protocol that is used for mobility support is shown in Figure 4-5. Messages (1) to (6) are used to setup the session between the Correspondent Host and the Mobile Node. The messages (7) and (8) accomplish the Load Control resource reservation for the upstream direction. For the downstream direction the messages (9) and (10) accomplish the Load Control resource reservation. Subsequently, the IP user data transfer in both directions, upstream and downstream, takes place. When one of the end terminals wishes to terminate the application then a combination of resource release and a session release procedure will be accomplished. Messages (11) to (14) are not used to refresh the resources that are already reserved. In Load Control this is one method of releasing the reserved resources. Therefore, the unit of resources on both the upstream and downstream directions that are not refreshed are released. Messages (15) to (17) are used for session release.

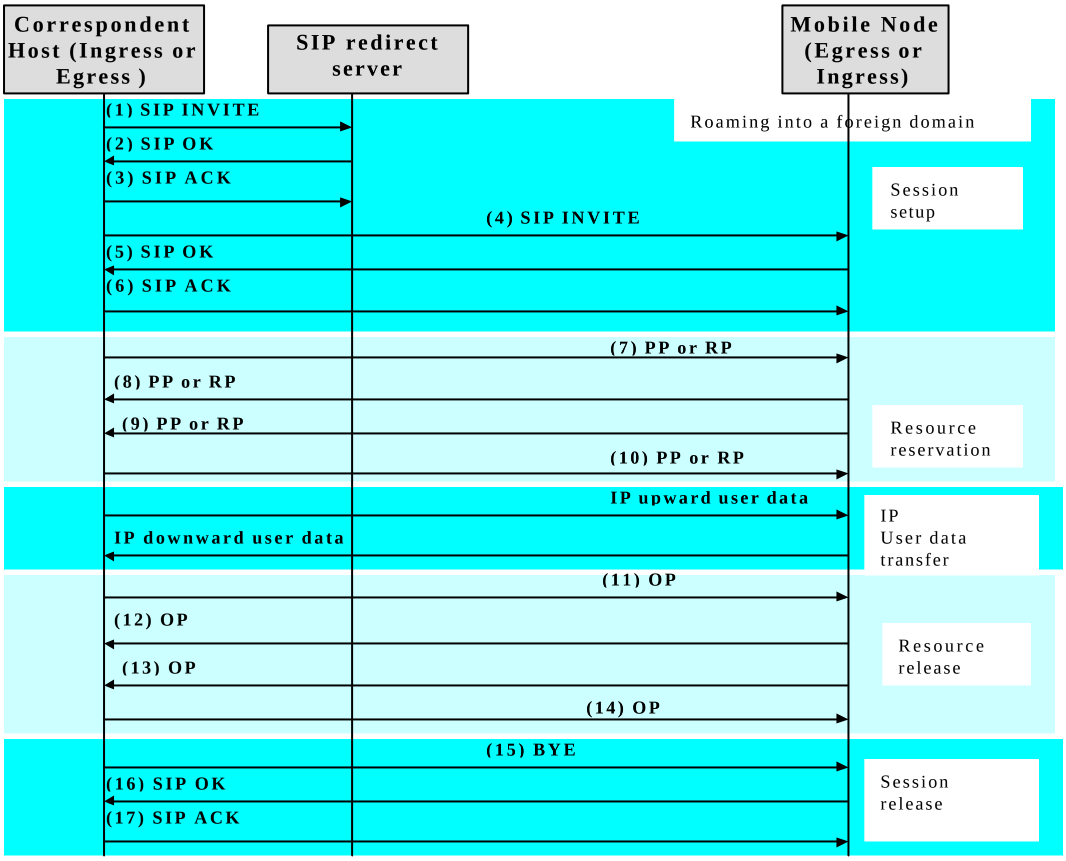


Figure 4-5: Interoperability between SIP (used for mobility support) and Load Control

4.6 Load Control and RSVP aggregation

Figure 4-6 views a possible interoperability scenario between the Load Control and the RSVP aggregation protocols.

In this scenario it is assumed that the combination of the RSVP aggregation concept viewed in Figure 2-5 and the Load Control protocol explained in Section 2.1.5 is used.

Messages (1) to (7) and the message (18) are specific RSVP aggregation messages that are used for the situation that a resizing of the RSVP aggregated states is required. The messages (9) to (11) and the messages (14) to (16) are specific Load Control messages that are used to reserve resources on all the routers that are active in the end to end communication and are located between the Aggregator and Deaggregator.

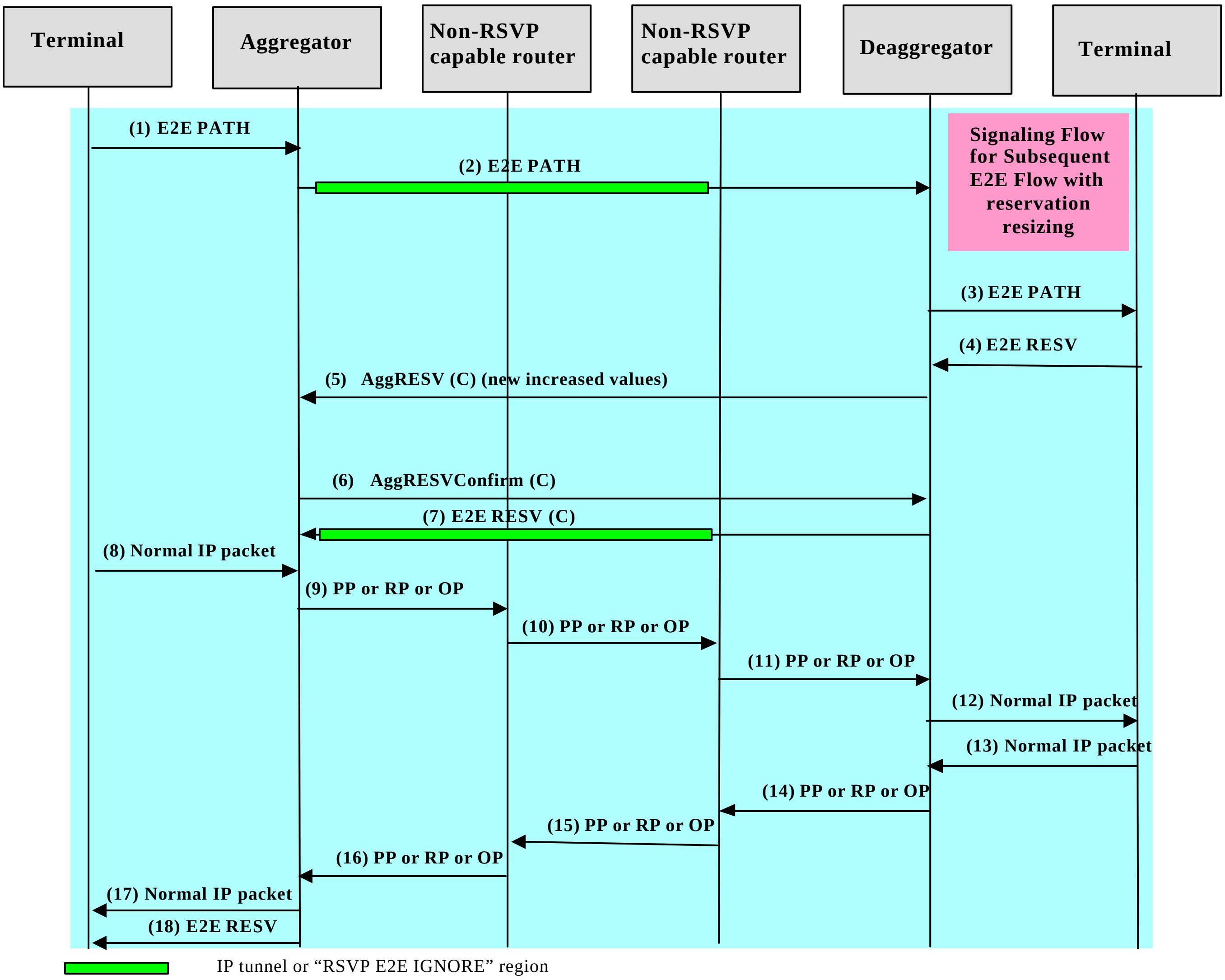


Figure 4-6: Example of an interoperability scenario between Load Control and RSVP aggregation

5 QoS & Mobility Framework Architecture Operation - an Exemplification

Specific realisations of the framework architecture will depend on the QoS architectures used at the *Access Network*s, management of mobility support, and related protocols.

The operation of this framework architecture can be described as consisting of certain procedures, that can be performed either sequentially or simultaneously, depending on the specific realisation of the framework:

- QoS “Session setup”: a session is initiated between the end hosts that are willing to start an application.
- QoS “Resource reservation”: reservation of the required resources in the access and / or core network.
- “IP user data transfer”: the flow of IP user data traffic.
- QoS “Resource release”: the reserved resources are released.
- QoS “Session termination”: the session is terminated.
- Network attachment: a mobile host attaches to a certain network, using a specific access technology.
- Network detachment: a mobile host detaches from a network

In this section we exemplify the operation of the QoS and mobility framework (see Figure 3-2). The detailed flow diagrams used to describe the functionality of the protocols used in this Section are described in the previous Sections 2, 4 and also in [KaRe00]. First, we give an example for the start of the communication, i.e. *session setup*, *resource reservation*, and *IP user data transfer*. Thereafter, we give an example of a hand-over from one access technology to another, using *network attachment*, *network detachment*, *resource release*, and *resource reservation*.

5.1 Start of Communication

The QoS “Session setup” and QoS “Resource reservation” procedures can be accomplished either simultaneously or sequentially. The same holds for the QoS “Resource release” and QoS “Session Release” procedures. Moreover, the QoS provisioning in the core network can be accomplished either on a static way (see Figure 5-1) or on a dynamic way (see Figure 5-2). The main difference in the operation of these two types of scenarios is that in the static QoS provisioning, the resource reservation and resource release is only accomplished locally at the access networks. In the dynamic QoS provisioning the resource reservation and release is additionally accomplished in the core network by using similar procedures as described in Section 4.6 and [KaRe00]. In [KaRe00] the Dynamic QoS provisioning is achieved by using a mixture of concepts, e.g., Bandwidth Broker, RSVP aggregation and Load Control.

For this particular example we assumed the following: the first five procedures, described above in Section 5, are performed sequentially. Moreover, any of the methods for QoS provisioning in the core network depicted in Figure 5-1 and Figure 5-2 are explained in Section 2, Section 4 and in [KaRe00]. It is important to note that during the network attachment procedure the QoS “Session Setup” procedure is applied only for the applications that are using the MDLG QoS mobility class. This is due to the fact that an MDLG QoS mobility class is supporting non-adaptive applications and therefore, the end application clients will need to agree on a deterioration of the provided QoS. The MDA QoS mobility class is supporting adaptive applications. This implies that the end application clients will always agree on a deterioration of the provided QoS.

The calling user, i.e. Host X is already attached to an *Access Network* supporting the Intserv QoS architecture and RSVP. It is attached to this network in two ways: using Bluetooth (on his home subnetwork), and using the GPRS access technology (on another subnetwork). The Host Y is also residing in an *Access Network* supporting the Intserv architecture. The *Application Client* in the Host X and Host Y is VoIP, i.e. non-adaptive with hard QoS requirements belonging to the Mobility Dependent Locally Guaranteed (MDLG) service class. Mobile IP manages the mobility support and *TS* at the host decides on the access technology. Host X and Host Y use SIP as *Session Layer Negotiation* protocol and RSVP enhancements as *End-to-End QoS Resource Reservation* protocol. *Application Clients* can be seen as SIP clients also.

QoS “Session Setup”

(S1): A calling user, *Application Client* in Host X, starts up a VoIP session to communicate with the called user – *Application Client* in Host Y.

(S2): At Host X, the *QoS API,* based on application attributes and QoS requirements, determines the Mobility Depend Locally Guaranteed (MDLG) service profile and translates these parameters in an understandable form for the underlying entities. Since, Host X is able to support more than one access technology, by using the technology selection procedure described in Section 3.3.2, *TS* will select the access technology that satisfies the predefined technology selection criteria for MDLG. Suppose for now, Bluetooth is selected.

(S3): By means of a SIP message, the calling user invites the called user, i.e. Host Y to start a VoIP session. The session description in the SIP message will contain the session name, purpose, media and timing information, and additional information regarding the bandwidth to be used by the VoIP application.

(S4): The called user, i.e. Host Y will perform the same procedures as Host X in ***(S2)*** and it will inform the calling user about the successful session setup completion, if the session is acceptable and the resources for the session are available in the remote access network.

QoS “Resource reservation”

(S5): Since both *Access Networks* support the RSVP/Intserv concept, then the calling user, i.e. RC in Host X, must start the “resource reservation” procedure (sending RSVP PATH messages).

In this procedure, the *RM* entities located in the *Access Networks* and *Diffserv Core Network* and the RC entity located in the Host Y will have to interoperate in order to reserve resources negotiated at the session layer. Preferably, in the *Diffserv Core Network* RSVP enhancements are used like RSVP aggregation or tunnelling to avoid scalability problems.

“IP user data traffic phase”

(S6) After successful completion of the QoS “session setup” and the QoS “resource reservation” procedure Host X and Host Y may start sending IP user data traffic, i.e. VoIP speech data.

5.2 Handover

If Host X gets out of the coverage of its home Bluetooth network, it has to rely on the GPRS network to continue the session. So, the assumption here is that Host X performs a handover from the Bluetooth subnetwork to the GPRS subnetwork during the exchange of data traffic, in order to remain connected to the Intserv-based access network. This will be handled by the *MC* and *MA* entities, in conjunction with the *RC* and *RM* entities. The specific example we give here exemplifies a so-called hard handover i.e. the old link is broken down before the new link is established.

Network detachment / Resource release

Possibly, the network detachment will be performed automatically, because Host X will loose contact with the Bluetooth network, and the soft state for Mobile IP and RSVP will be removed from the network. Alternatively, the state in the network is removed because it is being replaced by a new state, because of network attachment.

Network attachment

(Si-1): The *MC* entity of the Host X will try to find out from *MA* in the GPRS subnetwork what its new identity, i.e. IP address, is. In Mobile IP this is known as Care-of Address discovery.

(Si-2): The *MC* will send its new IP address to the *MA* in its home subnetwork, i.e. it will register. From now on, all IP packets will be tunnelled to the new IP address.

Resource reservation

(Si-3) Host X and Host Y will keep the current the QoS session setup and will re-initiate the “resource reservation”. The resource reservation will be done using the same QoS requirements as before. If the reservation is not successful, the QoS requirements will be renegotiated with the application. This may lead to either successful reservation or session termination.

(Si-4) After successful “ resource reservation” data exchanges follow as in ***(S6)***.

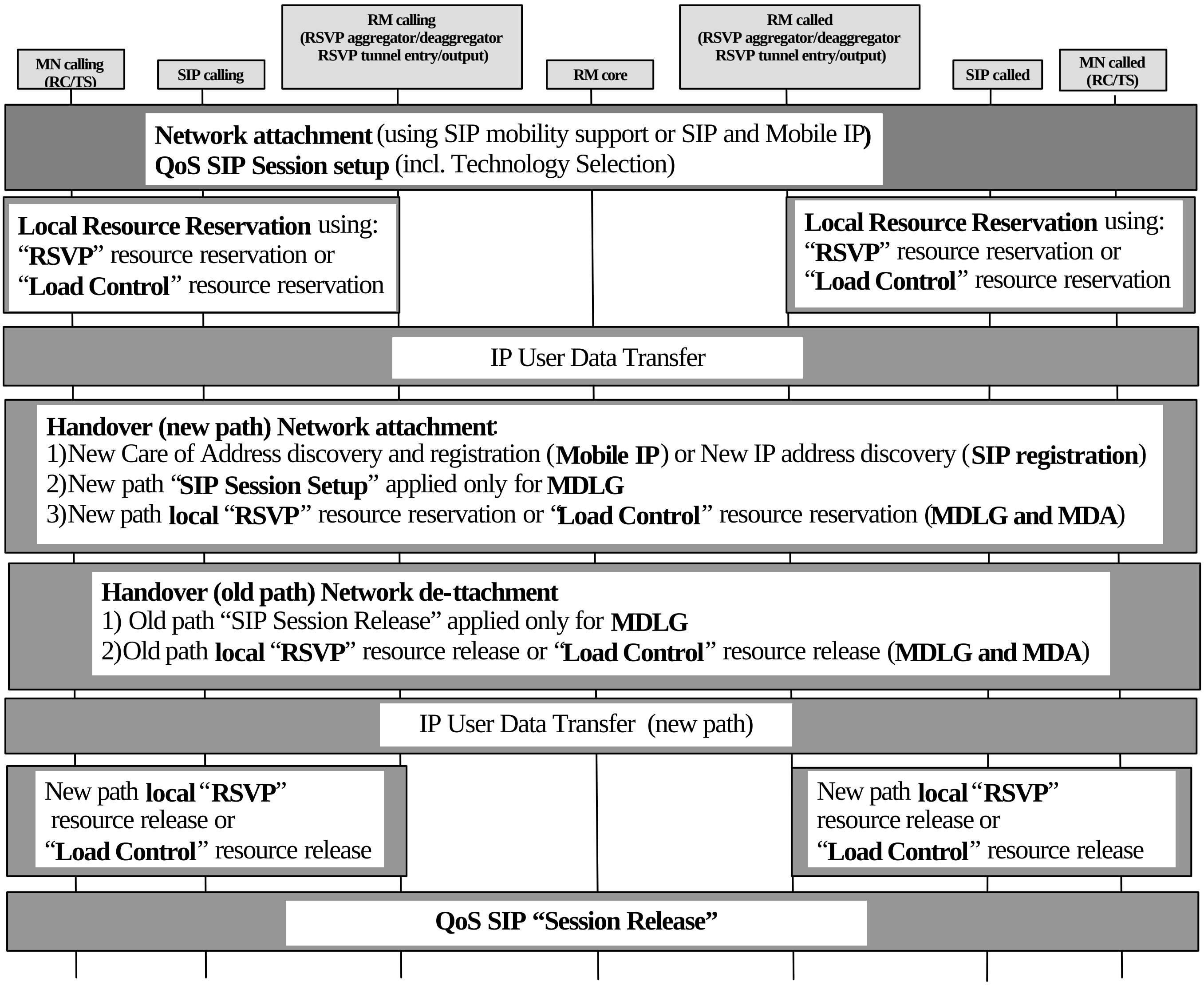


Figure 5-1: QoS & Mobility Framework Architecture Operation using static QoS provisioning in the core network

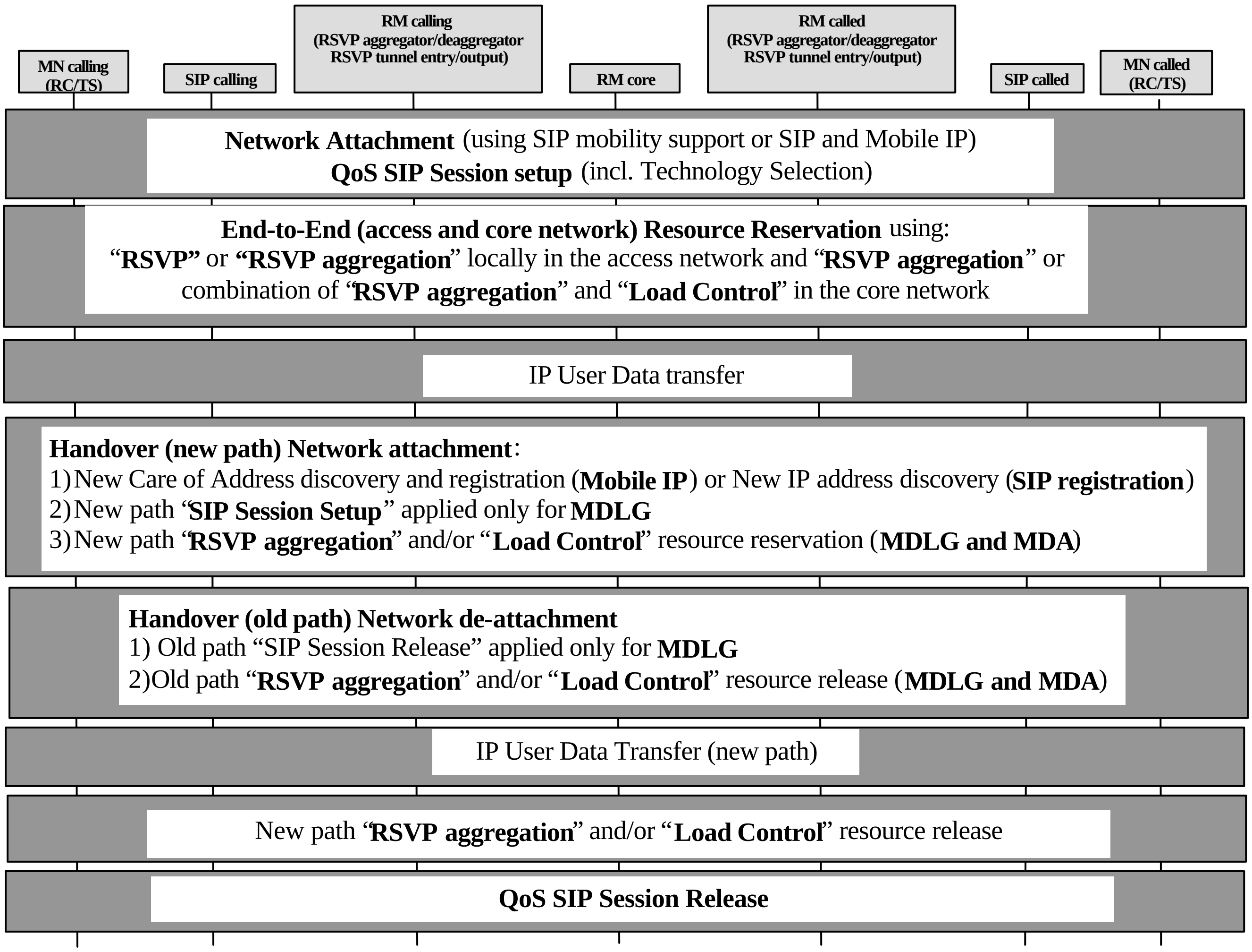


Figure 5-2: QoS & Mobility Framework Architecture Operation using dynamic QoS provisioning in the core network

6 Conclusions

In this document we have presented a new framework for the next generation Internet architecture that integrates QoS and mobility. This framework is capable of integrating various wired and wireless access and core technologies that are using different QoS architectures and protocols. The different QoS architectures located in the access networks can use a Diffserv core network to intercommunicate and provide end to end QoS support. The main advantages provided by this framework are related to the possibility of the session layer negotiation of QoS parameters before the actual network resource reservation procedures take place. This will enhance the scalability of the Diffserv core network and it will reduce the waste of resources in the access networks. Furthermore, the framework provides an efficient way of integrating the existing IP mobility protocols, such as the Mobile IP and resource reservation protocols, such as the RSVP and Load Control. Further work includes experimenting with combined RSVP/Intserv, Diffserv, Mobile IP and SIP implementations to demonstrate the feasibility of the framework.

7 Acknowledgements

Special thanks to Geert Heijenk, Martin van der Zee, Simon Oosthoek, and Rachid Ait Yaiz for their comments and suggestions, and to Phil Chimento and John de Waal for support.

8 References

[3GPP23.107] 3GPP technical Specification Group Services and System Aspects, "QoS Concept and Architecture, Release 1999, March 2000.

[BaIt00] Baker, F., Iturralde, C. Le Faucher, F., Davie, B., "Aggregation of RSVP for Ipv4 and Ipv6 Reservations", draft-ietf-issl-rsvp-aggr-02.txt, March 2000 (work in progress)

[BeBi99] Bernet, Y., Binder, J., Blake, S., Carlson, M., Davies, E., Ohlman, B., Werma, D., Wang, Zh., Weiss, W., "A framework for Differentiated Services", draft-ietf-diffserv-framework-02.txt, February 1999.

[BeDa94] Besse, L., Dairaine, L., Fedaoui, L., Tawbi, W., Thai, K., "Towards an Architecture for Distributed Multimedia Application Support", Proc., International Conference on Multimedia Computing and Systems, Boston, May 1994.

[BeYa00] Bernet, Y., Yavatkar, R., Ford, P., Baker, F., Zhang, L., Speer, M., Braden, R., Davie, B., Felstaine, E. "Framework for Integrated Services operation over Diffserv Networks", draft-ietf-issll-diffserv-rsvp-04.txt, March 2000 (work in progress)

[CaCo93] Campbell, A., Coulson, Garcia, F., G., Hutchison, D., Leopold, H., "Integrated Quality of Service for Multimedia Communications", Proc. IEEE INFOCOM'93, San Francisco, USA, pp. 732-739, April 1993.

[DuYa99] Durham, D., Yavatkar, R., "Inside the Internet's Resource reSerVation Protocol: Foundations for Quality of Service", John Willey & Sons, Inc., 1999.

[ErOh00] Eriksson, A., Ohlman, B., "A framework for On-demand QoS Bearers over Fixed and Mobile Internet Access", paper submitted to the IWQoS 2000, June 5-7, 2000.

[ISO95] ISO, "Quality of Service framework", ISO/IEC JTC1/SC21/WG1 N9680, International Standards Organisation, UK, 1995.

[Kar99] Karagiannis, G., "Mobile IP: State of the Art", Internet Next Generation report, located at "http://w3-emn.ericsson.se/project_Q_Wing/documents/mobip_a.pdf", 1999.

[KaRe00] Karagiannis G., Rexhepi, V., et. al., "QoS dynamic provisioning using Bandwidth Brokers, Trunk Reservations and Load Control", to be published in 2000.

[LaBh94] Lazar, A. A., Bhonsle, S., Lim, K.S., "A Binding Architecture for multimedia Networks", Proc., of COST-237 Conference on Multimedia Transport and Teleservices, Vienna, Austria, 1994.

[NaSm95] Nahrstedt, K., Smith, J., “Design, Implementation and Experiences of the OMEGA End Point Architecture”, Technical Report (MS-CIS-95-22), University of Pensylvania, May 1995.

[PeJo00] Perkins, C., Johnson, B., J., “Route Optimisation in Mobile IP”, Internet draft, draft-ietf-mobileip-optim-09.txt, Work in progress, February 2000.

[Per97] Perkins, C., E., “Mobile IP”, IEEE Communications Magazine, May 1997.

[Per98] Perkins, C., E., “Mobile networking through mobile IP”, IEEE Internet Computing, 1998.

[RFC1256] Deering, S., (ed.), “ICMP Router Discovery Messages”, RFC 1256, August 1989.

[RFC1633] Braden, R., Clark, D., Shenker, S., “Integrated Services in the Internet Architecture: An Overview”, IETF RFC 1633, 1994.

[RFC1905] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, 1996.

[RFC2002] Perkins, C., E., “(ed.) “IP Mobility Support”, RFC2002, proposed standard. IETF Mobile IP Working Group, Oct., 1996.

[RFC2003] Perkins, C., “IP encapsulation within IP”, RFC2003, October 1996.

[RFC2205] Braden, R., Zhang, L., Berson, S., Herzog, S., Jamin, S., “ Resource Reservation Protocol (RSVP) Version 1 Functional Specification”, IETF RFC 2205, 1997.

[RFC2210] Wroclawski, J., “ The use of RSVP with IETF integrated Services”, IETF RFC 2210, 1997.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Zh., Weiss, W., “An architecture for Differentiated Services”, IETF RFC 2475, 1998.

[RFC2327] Handley, M. and V. Jacobson, "SDP: session description protocol", RFC 2327, 1998.

[RFC2543] Handley, M., Schulzrinne, H., Schooler, E., Rosenberg, J., “ SIP: Session Initiation Protocol”, IETF RFC 2543, 1999.

[RFC2597] Heinanen, J., Baker, F., Weiss, W., Wroclawski, J., “Assured Forwarding PHB group”, IETF RFC 2597, 1999.

[RFC2598] Jacobson, V., Nichols, K., Poduri, K., “An Expedited Forwarding PHB”, IETF RFC 2598, 1999.

[RFC2638] Nichols, K., Jacobson, V., Zhang, L., “ A two-bit Differentiated Services Architecture for the Internet”, IETF RFC 2638, 1999.

[RFC2746] Terzis, A. Krawczyk, J., Wroclawski, J., Zhang, L., “RSVP operation over IP Tunnels”, IETF RFC2746, January 2000.

[RFC2748] Durham, D., Boyle, J., Cohen, R., Herzog, S., Raja, R., Sastry, A., "The COPS (Common Open Policy Service) Protocol ", IETF RFC, January 2000.

[RoVi00] Rosen. E., Viswanathan, A., Callon, R., "Multiprotocol Label Switching Architecture", IETF draft, draft-ietf-mpls-arch-06.txt, Work in progress, February 2000.

[Stev97] Stevens W. R., "TCP/IP Illustrated, Volume 1 The protocols", Addison Wesley Professional Computing Series, 1997.

[TeCh99] Teitelbaum, B., Chimento, P. "Qbone Bandwidth Broker Architecture" at "http://qbone.ctit.utwente.nl/deliverables/1999/d2/bboutline2.html" (work in progress)

[VoWo96] Volg, C., Wolf, L., Herrtwich, R., Wittig, H., "HeiRAT – Quality of Service Management for Distributed Multimedia Systems", Multimedia Systems Journal, 1996.

[WeSc99] Wedlund, E., Schulzrine, H., "Mobility support using SIP", ", Second ACM/IEEE International Conference on Wireless and Mobile Multimedia (WoWMoM'99), Seattle, Washington, August, 1999

[WeTu00] Westberg, L., Turanyi, Z. R.,.Partain, D., "Load Control of Real-Time Traffic", IETF draft, draft-westberg-loadcntr-03.txt, Work in progress, April 2000.

[Will99] Williams, B., "Quality of Service- Differentiated services and Multiprotocol Label Switching", Ericsson Internal White paper, August 1999.

[XiHa00] Xiao, X., Hannan, A., Bailey, B., "Traffic Engineering with MPLS in the Internet", IEEE Network, March/April 2000.

[ZeAi99] Zee van der M., Ait Yaiz R., "Quality of Service over Specific Link Layers" QWING State of the Art Report, located at: http://w3-emn.ericsson.se/project_ING/INGpublic/, 1999.

9 Abbreviations

AB	Access Boundary
AF	Assured Forwarding
API	Application Program Interface
BB	Bandwidth Broker
BR	Border Router
DB	Diffserv Boundary
Diffserv	Differentiated Services
DS	Differentiated Services

DSCP	DS Code Point
E2E	End to End
EF	Expedited Forwarding
ER	Edge Router
FA	Foreign Agent
GPRS	General Packet Radio Service
HA	Home Agent
ICMP	Internet Control Message Protocol
Intserv	Integrated Services
IP	Internet Protocol
ISP	Internet Service Provider
LSP	Label Switching Path
LSR	Label Switching Router
MA	Mobility Agent
MC	Mobility Client
MDA	Mobility Dependent Adaptive
MDLG	Mobility Dependent Locally Guaranteed
MN	Mobile Node
MP	Marked Packet
MPLS	MultiProtocol Label Switching
QoS	Quality of Service
OP	Ordinary Packet
PHB	Per Hop Behaviour
PP	Probe Packet
PSTN	Public Switched Telephone Network
RC	Resource Client

RM	Resource Manager
RP	Refresh Packet
RSVP	Resource reSerVation Protocol
SDP	Session Description Protocol
SIP	Service Initiation Protocol
SLA	Service Level Agreement
SLS	Service Level Specification
SMS	Short Message Service
TCP	Transport Control Protocol
TOS	Type of Service
TS	Technology Selector
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
W-CDMA	Wideband - Code Divission Multiple Access
W-LAN	Wireless- Local Area Network