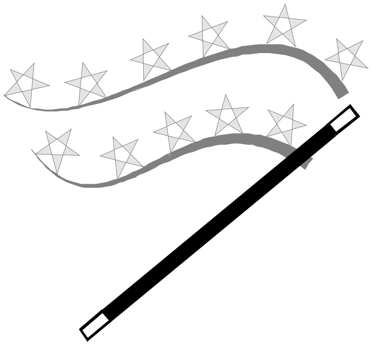
ACTS Project AC085

Wireless ATM Network

Demonstrator

The Magic WAND



Deliverable 1D6
IP Over Wireless ATM

DELIVERABLES

Part I

Project Number:	AC085
Project Title:	The Magic WAND
Deliverable Type (P/R/L/I):	P

CEC Deliverable Number:	AC085/NMP/R&T/DR/P/032/b1
Contractual date if delivery to the CEC:	31.08.1998
Actual date of delivery to the CEC:	31.08.1998
Title of deliverable:	IP over Wireless ATM
Workpackage contributing to the deliverable:	WP1
Nature of the deliverable (P/R/S/T/O):	R
Author(s):	Lorraine Stacey, Juha Ala-Laurila, Jouni Mikkonen, Jukka Seppälä, Stathes Hadjiefthymiades, Neda Nikaein, George Fankhauser, Sarantis Paskalis

Abstract:	The rapid evolution of Internet services has created a strong need for broadband networks with high data rates and support for quality of service, which users can access in a wireless environment. This deliverable describes how the WAND broadband radio technology can be modified to support best effort and real-time unicast and multicast Internet applications in a wireless environment. The document proposes an IPv6 based network architecture in which time critical IP packet streams are transmitted with higher QoS priorities. Current WAND control signalling is replaced with a light-weight alternative. By default all traffic is carried as best effort data. A mobility enhanced router is described which identifies IP flows that should receive higher priority service, and marks the packets accordingly. This enables the radio link to provide the appropriate QoS to each packet stream. The proposed system supports both unicast and multicast applications, via a centralised multicast group management function performed by the mobility enhanced router. The proposed wireless broadband IP system offers mobility both within a wireless IP subnet and between different IP subnets.
Keyword list:	

Part II

(Executive Summary)

The rapid evolution of Internet services has created a strong need for broadband networks with high data rates and support for Quality of Service (QoS). Users want to access these services in the wireless environment. Two main obstacles exist: Current wireless networks do not provide sufficient QoS support. Furthermore, existing wireless networks are not capable of serving several simultaneous connections with high data rates. To meet the increased user demands, new wireless broadband network techniques have to be researched and developed.

Wireless access networks are being enhanced to provide mobile broadband real-time services. Given ATM is able to support applications with a range of service requirements, there has been strong interest in developing wireless ATM technologies. One benefit of ATM in the broadband wireless environment is the tagged, small fixed size nature of ATM cells which is well suited to high speed radio interfaces. The Magic WAND (Wireless ATM Network Demonstrator) was originally designed to provide a high speed ATM radio sub-system that maintains QoS guarantees. The current WAND demonstrator is intended to support two classes of application: best-effort Internet applications, and real-time native ATM applications.

The global trend is towards QoS aware IP networks. Instead of being a state-of-the-art LAN technology, ATM seems to be gaining widespread use as a backbone technology only. Therefore, the WAND project also studies how to modify the developed broadband radio technology to support real-time Internet applications in a wireless environment.

This deliverable describes the results of the WP1 extension "*IP over Wireless ATM*" which is defined in the Technical Annex. This document proposes an *IPv6* based network architecture in which time-critical IP flows are transmitted with higher QoS priorities. The idea is to employ flow identifiers which *map radio flows and IP flows together and allow the user to assign different IP QoS parameters for various flow types*. The current WAND *control signalling* is replaced with *an Ipsilon Flow Management Protocol (IFMP) like protocol*. This is used for classifying IP flows in the core network interface (i.e. in IP routers) and for managing flow identifiers. By default all IP packets are carried as best-effort data. As the mobility enhanced IP router detects an IP flow it marks the packets belonging to that flow with a dedicated flow identifier. This enables the radio link to support IP packets with varying IP QoS parameters properly. Support for both Integrated and Differentiated services is provided. In addition both unicast and multicast applications are supported. Multicast group management is provided in a centralised manner by the mobility enhanced router. Finally, the proposed wireless broadband IP system offers mobility both within a wireless IP subnet and between different IP subnets.

Part III

Scope of the Document

The objective of the document is to define a system architecture for the indoor wireless broadband network which enables native Internet Protocol (IP) access to both conventional narrow-band packet data services and to delay-sensitive broadband multimedia services. The work covers the entire network including the mobile terminal, the radio access network and the switching backbone.

The implemented wireless ATM access system (target WAND) is considered as a starting reference that has been modified according to the requirements of the IP architecture. Briefly the requirements are as follows:

- to transport both unicast and multicast native IP traffic,
- to meet the Quality of Service requirements of both traditional best effort IP traffic and real time IP traffic, and
- mobility support for a mobile terminal both within the access network and between access networks.

This deliverable *defines a system architecture for an indoor wireless broadband IP network* concept which meets the above mentioned requirements. The document is divided into two sections:

The first section describes the overall system architecture and design framework for a wireless IP access network. This part describes alternative approaches for implementing the system components. The objective is to provide relevant background information and to propose design guidelines.

The remainder of the document contains a detailed system specification covering connection management, QoS management, multicast support and mobility aspects. This part defines the suggested system architecture and the key functionality. Annex A includes the specific Message Sequence Charts (MSCs) of the most important functional scenarios.

Section I: Design Framework

1. INTRODUCTION TO "IP OVER WIRELESS ATM"

In recent years, the use and variety of Internet applications has grown enormously. Traditionally, the Internet supported text based applications including e-mail, telnet and the File Transfer Protocol (FTP). Now, many applications involve audio, video, images and text (e.g. World Wide Web (WWW), and multi-player games). In addition, many of these applications have strict performance requirements (e.g. Internet telephony and multi-player games). In response, the Internet is moving from a best effort model to a system, capable of supporting a range of traffic characteristics and service requirements.

Asynchronous Transfer Mode (ATM), was initially developed to provide a single integrated network capable of supporting applications with a range of traffic characteristics and service requirements. Hence, ATM is a leading contender, when considering which network topologies best support the evolved Internet. Compared to many other network technologies, ATM has the additional advantage that, like the Internet, it can operate at a range of link rates over any distance and physical media. As a result, many researchers are investigating how to provide Internet services over ATM networks.

Simultaneously wireless access networks are being enhanced to provide mobile broadband real-time services. Given ATM is able to support applications with a range of service requirements, there has been much interest in developing wireless ATM technologies. Another benefit of ATM in the broadband wireless environment is that the tagged, small fixed size nature of ATM cells is well suited to high speed radio interfaces [Fre97]. The Magic WAND system was designed to provide a high speed ATM radio sub-system that maintains QoS guarantees. The WAND demonstrator, in its current form supports two classes of application [Mik97]: best effort Internet applications and real-time native ATM applications. However, the IP access is based on LAN Emulation (LANE) which can not differentiate various IP traffic flows, or provide QoS guarantees. Therefore a new native wireless IP concept has to be developed.

2. DESIGN GOALS

At the moment there is a strong global trend towards IP. Indeed the Internet is currently growing exponentially. Major problems with the current Internet are routing table size and computation, and IP address exhaustion [Dee95]. Therefore, a new IP version called IPv6 (or IPng) is being developed. Other IP development work has also introduced important mobility and QoS features suitable for both IPv4 and IPv6. These features will help IP to become a mobile multimedia capable networking technology for all services. Several important Telecommunication manufacturers, such as Microsoft, Ipsilon and Cisco, have indicated that they will implement IPv6 in their future products. Therefore, IPv6 was seen as a relevant starting point for the future wireless broadband system. New IPv6 features will provide integrated support for *terminal mobility*. These include built-in mobility with optimised routing, address auto-configuration, DHCPv6 (Dynamic Host Configuration Protocol) and IP security (IPSec) mechanisms. These features enhanced with appropriate wireless access network specific modifications will enable IP based mobility between IP subnets, secure communications with IPSec and provide registration support mechanisms for wireless IP networking. Due to the enhanced features offered by IPv6, the developed system architecture will be based on IPv6 rather than IPv4.

Another goal is to define a solution that maximises scalability and compatibility with the developed wireless 5GHz broadband radio (WAND radio). The wireless IP system has to be specified in a way that it can be implemented utilising the developed WAND radio sub-system.

The system should support state-of-the-art fixed network IP QoS mechanisms in a wireless environment. The most important IP QoS mechanisms are being developed by IETF QoS work groups. There are currently many different signalling/control protocols related to Internet connection/QoS management. For instance, RSVP (Resource Reservation Protocol) [Bra96], Differentiated Services [Dif] and application level session management protocols (e.g. Session Initiation Protocol [Han98]). The defined QoS Management scheme has to be compatible with existing core network QoS techniques. Furthermore a flexible mapping is needed from all of these protocols into the radio access QoS mechanisms.

Multicast applications such as video conferencing, computer supported collaborative work and multi-player games are becoming increasingly popular in the Internet. Hence future wireless networks must provide IP multicast support. This translates into a need for multicast group management mechanisms and a way to transmit traffic to multiple senders. The system also needs to inter-operate Internet multicast routing protocols.

Finally, the system must provide mobility support both within one wireless IP network and between wireless IP networks. Given the first design goal of basing the system on IPv6, mobility between systems must be based on IP protocols. Thus, the IP address of the mobile terminal will change as it moves between different networks. As a result inter-network mobility is based upon Mobile IP, and specifically the Mobile IP specification for IPv6.

3. SYSTEM CHARACTERISTICS

Quality of Service Support

The developed wireless IP system is capable of reliably delivering time-critical (multimedia) data streams, which calls for a proper QoS management scheme. In the fixed IP network all IP packets are typically sent as best effort data traffic in which all data streams are treated with equal priority, which may have a dramatic effect on multimedia services. Therefore, the system has to provide *a reliable mechanism for detecting and differentiating time-critical data flows from the best effort data*. The defined concept *explicitly assigns different QoS classes* (e.g. bandwidth, traffic class) to various IP packet flows. The wireless system is intended to be compatible with both IETF IP QoS approaches: *integrated and differentiated services*.

Mobility Management

The wireless network *supports mobility both within the IP sub-network* (intra sub-network mobility) *and between different IP sub-networks* (inter sub-network mobility). Due to the connectionless nature of the IP network the handovers are specified to be hard and *lossy*. The mobility management scheme covers both radio access network and core network. The radio access network mobility management is based on the handover scheme contributed to BRAN standardisation forum while the core network mobility utilises standard integrated IPv6 mobility features some of which were optimised for wireless environment.

User Authentication

In a private network terminal authentication alone may be adequate for security. However, in public networks both user and terminal identification are essential for billing and security. Therefore, the system identifies both the terminal devices and the users. The user authentication scheme involves *both radio access network and optional IP network level authentication*.

Multicast Support

The wireless IP network supports standard IP multicast and *optimises multicast performance in the radio access network*. The multicast mechanism is optimised for the wireless link. The access router handles multicast messages and transmits them only to the access points that have members of the particular multicast group in their coverage area. The access point sends multicast messages simultaneously to all multicast members using a specific radio level multicast address.

Compatibility

The radio access network can be connected to any standard fixed IP core network either using ATM or Ethernet links. The flow and QoS management functions are defined as extensions to the standard IPv6 protocol stack and the network mobility scheme conforms to the IPv6 mobility scheme. The defined terminal architecture is capable of running all de-facto IP services and applications over the radio link.

4. SYSTEM ARCHITECTURE

4.1 Physical Network Structure

4.1.1 Theoretical Reference model

The system architecture follows the outlines of the theoretical General Radio Access Network (GRAN) reference model, illustrated in Figure 1 [ES21]. The GRAN broadband radio access network reference model, defined in the Universal Mobile Telecommunications System (UMTS) concept, consists of a Radio Access Network (RAN), a Broadband Radio Access Network (BRAN) and different core networks. The BRAN includes the RAN and core network dependent inter-working functionality (IWF) blocks. The RAN covers all of the radio dependent parts and the IWFs link the RAN to various core networks and terminal entities. Hereafter the term *wireless IP system* refers to the *entire BRAN network covering RAN and necessary IWFs*. The wireless IP system is connected to the IP core network.

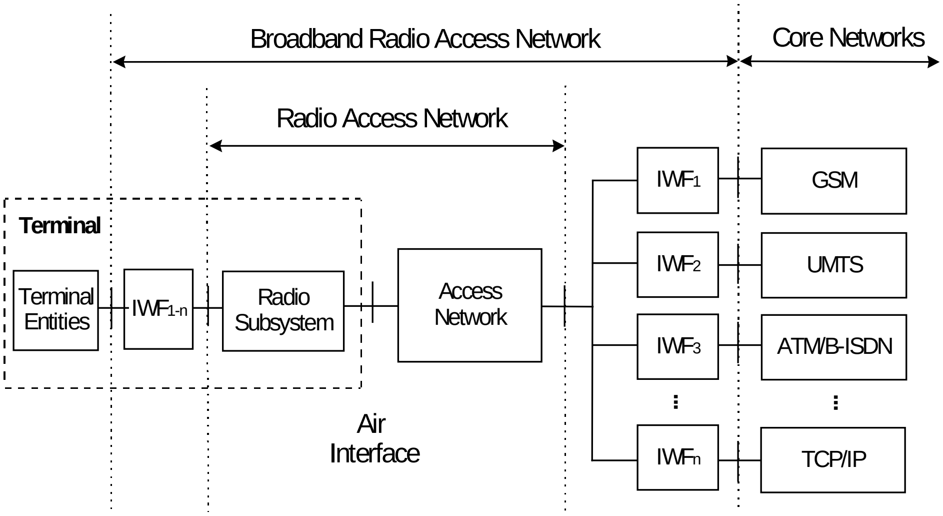


Figure 1: General GRAN Reference Model [ES21]

4.2 Network Entities

The network, depicted in Figure 2, is composed of Mobile Terminals (MT), Access Points (AP), and mobility enhanced IP routers (termed M-router or MR).

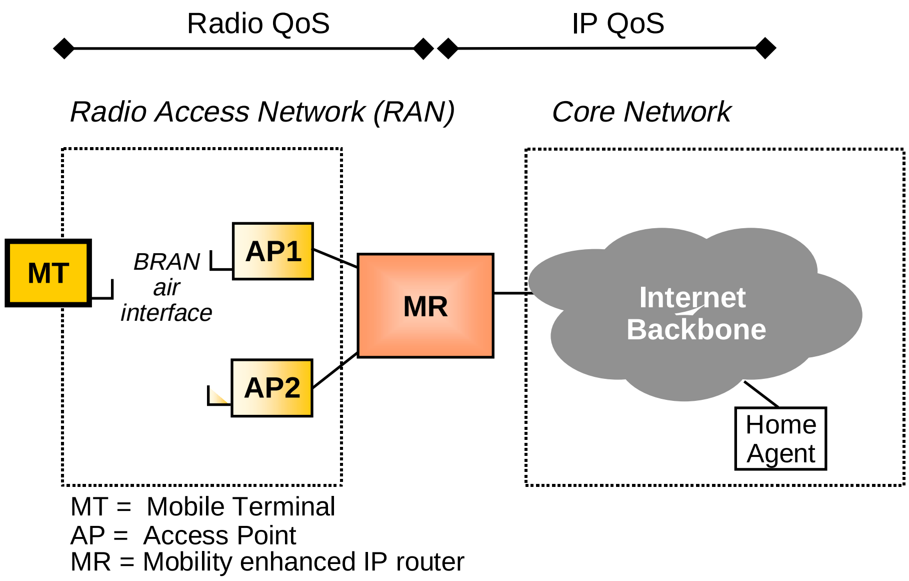


Figure 2: General System Architecture

Radio Access Network (RAN) implements all the radio dependent functionality such as radio resource management, set-up and release of wireless flows, handovers and packet compression. RAN contains MTs and APs.

Mobile Terminal (MT) is the user device for accessing wireless Internet services. The terminal will be the end point of the Internet and radio access network control protocols.

The Access Point (AP) implements all of the radio dependent control functionality, such as radio resource management. APs include radio resource management and radio link control functions. The corresponding network elements in GSM are BTS/TRX and BSC.

M-router (MR) is the boundary of the wireless IP subnetwork and manages one or more access points. The M-router handles the mobility and location management of terminals that are registered to access points. It will provide IP mobility services and address allocation functions. IP address auto-configuration is used to allocate IP addresses to mobile terminals. DHCP can also be used to manage IP addresses for the terminals. The benefit of DHCP (compared to address auto-configuration) is that it can also be used to implement security policies concerning the allocation of and access to new local addresses. The corresponding element in the GSM/GPRS network is the GGSN.

Home Agent (HA) - IP home agent functionality is needed to provide mobility between wireless IP subnets. Note that the functions of the home agent are as in standard Mobile IP. The HA resides in the home network of the MT and is accessed through standard IP gateways. Typically the HA is implemented as part of the MR of the home network. However it can also be a separate entity. Potentially the HA could be extended to contain user authentication information and a billing database (assuming a one-to-one association between user and MT). In this case it begins to resemble a GSM HLR.

Reference points for interoperability are assumed to exist in three places:

- radio link interface (MT - AP)
- core network interface (AP - MR)
- terminal - core network interface (MT - MR)

The radio interface is assumed to follow the BRAN HIPERLAN/2 standardisation. The standardisation of the AP-MR interface will allow users to purchase their radio access network from any manufacturer. The MT-MR interface is a logical interface. Standardisation of this (e.g. radio flow signalling) will allow the same MR device to be accessed by different types of terminal.

4.3 Target Environment

The WAND system is targeted for private and public networks. Public networks are typically operated by Internet service providers or telecom operators. Common places for public wireless access networks are hot spots, such as airports, hotels, railway stations etc. In this case the public network operator has to be able to reliably authenticate the users for billing purposes. In addition the network should offer security on the IP level.

Business LANs provide another interesting application area for the WAND system. Here the WAND system provides a wireless extension to the existing fixed LAN infrastructure. Typical company LANs are based on Ethernet cabling. Therefore, the access points are assumed to be connected to the M-router via Ethernet. The M-router can serve both fixed terminals and the mobile terminals.

5. QUALITY OF SERVICE MANAGEMENT

5.1 Internet QoS

Internet QoS can be expressed as the combination of network imposed *delay*, *jitter*, *bandwidth* and *reliability* [Fer98]. Delay is the elapsed time for a packet to traverse from the source to the destination. Jitter is the perceived variation in end-to-end delay. Bandwidth is the maximal data transfer rate that can be sustained between source and destination. Finally, reliable transmission delivers all the packets in the correct order, without dropping them or causing bit errors. Reliability is a property of the transmission system and is affected by the average error rate of the medium and by routing/switching design. In the fixed Internet packet loss is caused mainly by congestion. However in wireless networks both congestion and the unreliability of media must considered.

When we assess QoS provision in the end-to-end network, traversed transmission links must be examined in terms of the four basic QoS metrics described above. ATM (Asynchronous Transfer Mode) is a solution relying on a homogenous network with a unified data unit, the ATM cell, and QoS guarantees are achieved with statistical multiplexing. While ATM is the state-of-the-art QoS platform, starting from the "bottom-up", it seems increasingly unlikely that complete end-to-end networks will be based on ATM. Instead IP will be the real connecting networking layer. IP operates over any data link e.g. ATM, Ethernet and Frame Relay, and over any physical link such as copper, optical fibre and wireless. Given the low probability of homogeneous end-to-end data link networks, QoS signalling should be IP driven and separated from the specific QoS mechanisms within one transmission link (see Figure 4).

The Internet has traditionally operated on a "best effort" basis. Best effort service has been adequate for legacy data traffic and applications that can adapt to bandwidth and delay variations. However, the type of applications carried over the Internet is expanding. As a result new end-to-end QoS requirements for the Internet backbone are being set for instance by voice and video service users.

IP QoS work is being conducted within the IETF. This work focuses on two main streams for QoS control, namely Integrated Services [She95],[Bra97] and Differentiated Services (or diffserv) [Dif]. RSVP is an end-to-end control protocol, forming the signalling part of the Integrated Services Architecture. It allows users to communicate their QoS requirements to routers on the data path. Thus RSVP requires (soft) connection state maintenance in each router along the data path. The disadvantages of RSVP relate to the scalability of Internet backbone operation [Fer98]. First of all, RSVP state maintenance in routers consumes a lot of memory. Secondly, periodic state refresh messages increase the Internet traffic load. Recently, it has been recognised that the integrated services architecture signalling protocol, RSVP, can interoperate with other QoS mechanisms, including native ATM or diffserv mechanisms [Ber98]. Using RSVP as an end-to-end signalling protocol is therefore independent of the employed QoS mechanisms of the core-network.

Differentiated Services specifies IP header bit usage to differentiate between QoS classes of service. Discussion within the IETF focuses on employing the IPv4 *Type of Service* and IPv6 *QoS Class* fields for this task [Fer97],[Jac97]. Proposals include varying the number of pre-defined bits for delay and dropping priorities to allow each packet to be handled independently based on the header bits. The main objective is to specify a QoS mechanism based solely on the contents of header fields rather than end-to-end flow management. The operation of RSVP and Differentiated Services are described in more detail below.

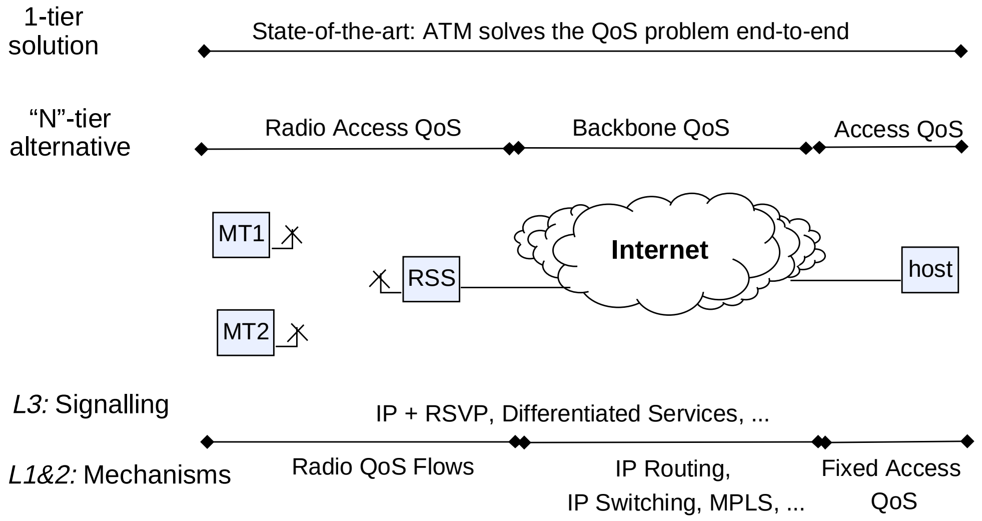


Figure 3: QoS Signalling in the Network

5.1.1 RSVP

To reserve resources across the network, RSVP uses the One Pass With Advertising (OPWA) mechanism. Senders "advertise" application traffic characteristics in PATH messages. Routers between senders and receivers modify PATH messages to describe the service they provide (e.g. how much delay they contribute to the overall end-to-end delay). Receivers determine their QoS requirements on the basis of the PATH message contents. The receiver QoS requirements are transmitted to the sender in RESV messages. The information contained in RESV messages is then used by routers and senders to reserve the requested QoS. This process is illustrated in Figure 8. Refer to [Bra97], [Man97].

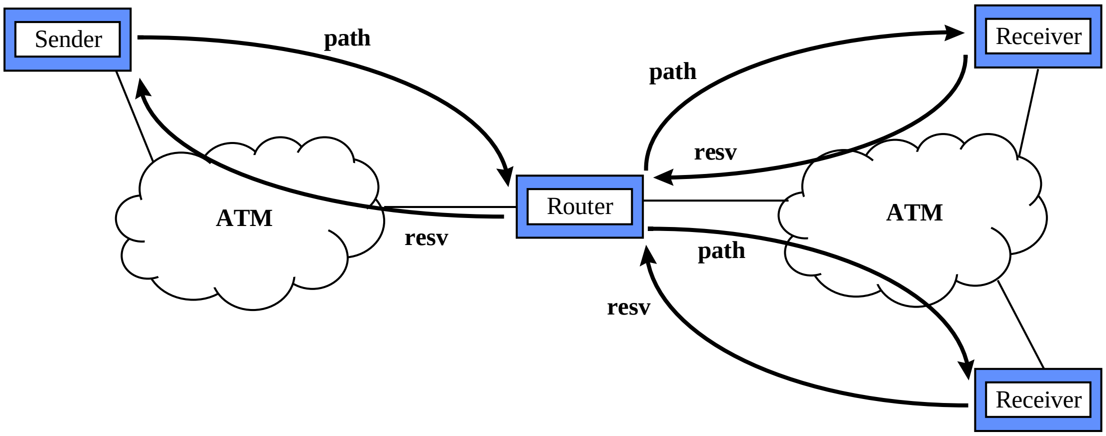


Figure 4: RSVP Operation

In RSVP the session specification combined with a filter specification (spec) defines the set of packets that will receive the reserved QoS. RSVP defines a session by its: Destination Address, IP Protocol ID and optionally the Destination Port. Each filter spec contains the Source Address, and optionally the Source Port. The filter specifications associated with a session depends on the reservation style chosen as described below.

In explicit sender selection reservation, each filter spec must match exactly one sender. Hence the resources are reserved for packets from a single sender to a set of receivers. In contrast the wildcard reservation style reserves resources for traffic from any sender to the session's set of receivers. This means that no filter spec is needed since the reservation is shared over all possible senders. The third alternative reservations style is shared sender selection where the reservation is shared by a specific set of senders. In this case, the reservation request contains a filter spec for each sender.

[Cra97],[Ber97a],[Ber97b] describe how to map RSVP to UNI 3.1 or UNI 4.0 ATM networks. The currently recommended approach transports RSVP messages via best effort VCs. Futhermore QoS data support is provided by either the limited heterogeneity model or the modified homogeneous model. In the limited heterogeneity model, two VCs are maintained for each reservation, one providing best effort service and the other the maximum resources requested by receivers. The modified homogeneous approach also creates a VC that reserves the maximum resources requested by receivers. However, in this case the best effort VC is only created when a receiver with best effort requirements is unable to join the QoS VC (e.g. because the bandwidth of the links to this receiver are insufficient.). This means the sender only replicates traffic (one copy for each VC) when some receivers can't support the maximum QoS VC.

An alternative aggregated approach has been proposed. This treats the ATM network as a series of pt-pt links each able to carry multiple Internet flows. This reduces the problem of supporting RSVP over ATM to a known problem since much of the existing Internet infrastructure comprises pt-pt links. Issues such as the resources to reserve for such aggregate VCs are being considered further by the ISSLL IETF working group.

A list of some routers available that do support RSVP according to the IETF RSVP survey, July 1997 are presented in Table 1.

Table 1 RSVP Support

Producer	**Model**[1]	**Other Details:**
3COM	CoreBuilder (status: alpha prototype) and NetBuilder II (status: development)	Both systems are based on ISI RSVPD reference software. CoreBuilder supports Guaranteed and Controlled Load Service while NetBuilder II supports the Controlled Load Service.
Bay Networks	Backbone Node, Access Stack Node, Advanced Remote Node, Access Node (status: beta prototype for all)	All models support the Controlled Load Service.
IBM	QoS-Switch-Router (QSR) (status: Beta testing. It will be used as a research prototype).	Not based on ISI RSVP reference software. Integrated services model supported: Guaranteed Service, Controlled Load (and Committed Rate). The hardware is based on IBM MSS Router and 8260 ATM switch. The operating system is IBM common router architecture (also called common code). The code is designed to operate either as a standard router implementation, or as an RSVP proxy agent that enables the router to make RSVP reservations on behalf of external IP flows. Specifically, as a router, it is capable of processing RSVP messages and of making the required resource reservations on each interface. At the same time, the proxy capability allows the router to initiate RSVP PATH/RESV messages to set-up RSVP flows across the network.
CISCO	IOS 11.2. Supported on all 1xxx, 25xx, 4xxx, and 7xxx platforms. Product since August 1996.	Not based on ISI RSVP reference software; independently implemented. Integrated services model supported: Guaranteed and Controlled Load. Capable of proxying for non-RSVP hosts
FORE	RSVP Prototype (status: alpha prototype)	Not based on ISI RSVPD reference software.
Furukawa Electric	INFONET3740/3780/3791 (status: beta prototype)	Based on ISI RSVPD reference software. Integrated services model supported: Controlled Load. Presently (July 1997), it doesn't support multicast routing; it supports RSVP only for unicast.
IBM	IBM Multi-protocol Router Family (Nways Router), including 2210, 2216, and 8210 (Status: under testing).	Integrated services model supported: Controlled Load. The platform's software has been developed in IBM Watson Research Lab's. Supports RSVP QoS traffic over ATM SVC links.

5.1.2 Differentiated Services

During the last year the IETF Differentiated Services (DS, or diffserv) WG has been developing a way to provide different levels of QoS without the need for signalling. It is believed that Internet Service Providers (ISPs) will not want to use signalling mechanisms such as RSVP to provide fine grained resource allocation. Instead users' needs can be satisfied by offering a range of different services from which they can select. Strict real time per flow guarantees are not envisioned as a goal of the diffserv effort, rather the provision of different levels of best effort service differentiated on timeliness of delivery and drop precedence. A low loss, low delay service will be priced higher than a service that provides worse performance in either of the two parameters.

[1] The status of products is as recorded in July 1997.

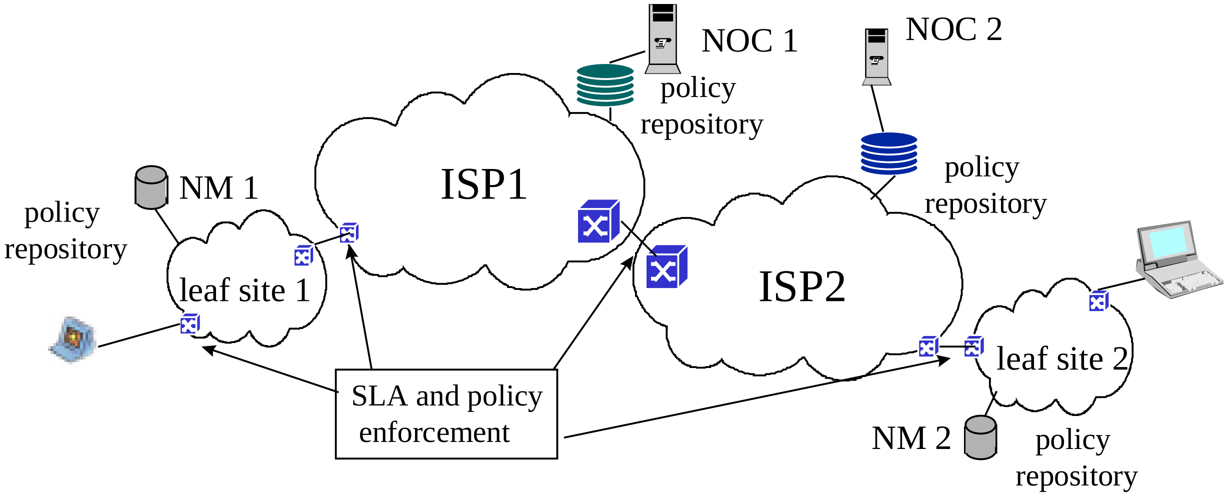


Figure 5: Differentiated Services Operation

The approach which is going to be used in the DS architecture is based on a set of building blocks that can be used to create a wide range of different services. The basic building block is the Per Hop Behaviour (PHB). Any packet will carry in the DS field of the IP header (bits 0 - 5 of the TOS and Traffic Class fields in the IPv4 and IPv6 headers respectively) a code point (that is a value) which will be used to select the proper packet handling that the router will have to provide in accordance to the definition of the PHB that the code point uniquely identifies. A configurable table with a code point to PHB mapping will be used in any router in order to allow flexibility and operator's custom needs. A packet has to be classified into proper queues based on the DS field value.

A service is defined based on a contract which specifies the characteristics of the traffic (e.g. by means of a token-bucket) the user is allowed to inject in the Provider's network and the expected service the contract-conforming traffic will experience. The traffic eligible for a certain service may be marked by the user with a specific value of the DS field. The ISP may remark the DS field based on internal policies or as a consequence of user's misbehaviour, detected by traffic policing. A TCA (Traffic Conditioning Agreement) may be used to specify such actions and values of the DS field used at the boundary. In the agreement some requirements such as traffic shaping may be included. The service is implemented by traffic conditioning and policy enforcement at boundaries (where the boundary may be a first hop router, an ISP access point, an inter-provider peering point etc.).

Network engineering and provisioning is fundamental for meeting service commitments successfully. Dynamic approaches to service provisioning based on a Bandwidth Broker, or possibly making use of RSVP, have been proposed in some papers or informational documents, but the IETF is not going to specify how to provision the network. Furthermore, some signalling may be used to negotiate resources dynamically (once again RSVP may be used) but at the moment this is out of the standardisation goals the WG has in its charter.

5.1.3 IP QoS Technique Independent QoS Management

To complicate issues further, the Internet also supports various session management protocols at the application level. The H.323 standard defined by the ITU (International Telecommunication Union) describes video conferencing over LANs and the Internet [H323]. The H.323 standard covers connection control for audio and video telephony, gateways for e.g. ISDN and POTS inter-operation, and gatekeepers for admission control and address translation. H.323 is a fairly complex and large standard [Lan97]. SIP (Session Initiation Protocol) [Han98], on the contrary, is a lightweight protocol designed to enable the invitation of users to participate in multimedia sessions. SIP may be used in conjunction with other call set-up and signalling protocols and it can invite users to sessions with or without resource reservation. Both H.323 and SIP carry Internet traffic related QoS information that provides useful information for radio access network QoS operation.

To summarise, there are many different signalling/control protocols related to the Internet connection/QoS management. Many of these are already designed to interwork with each other. A flexible mapping is needed from all of these into the radio access QoS mechanisms. Hence it makes sense to separate radio access QoS mechanisms from IP level QoS management. This also has the benefit that the radio access QoS mechanism is independent of the QoS mechanism employed in the core Internet.

5.2 QoS Management Scheme

All IP communication is packet based relying on connectionless transmission. Therefore, the system could deploy a light-weight control mechanism for managing IP flows, which reduces implementation complexity significantly compared e.g. to the wireless ATM system. Here the term *IP flow* refers to the flow of IP packets that belong to the same “connection”, i.e. the IP packets that are sent between particular applications (port) and between particular hosts (IP addresses). IPv6 includes a mechanism for assigning flow labels to IP packet flows. As discussed above, the IETF has developed several techniques for assigning IP level QoS parameters for various flows.

The wireless IP system has to be capable of distinguishing various IP flows both in the network side and in the air interface. In this scheme the M-router and radio sub-system can assign varying QoS parameters (such as scheduling priority and guaranteed bandwidth) for different IP flows. Furthermore, the defined BRAN (WAND) radio sub-system allows different QoS parameters to be allocated to various connection types. Thus it is essential to specify a mechanism for mapping IP level QoS parameters assigned to an IP flow into BRAN radio flow QoS. Figure 6 illustrates the principles of QoS mapping.

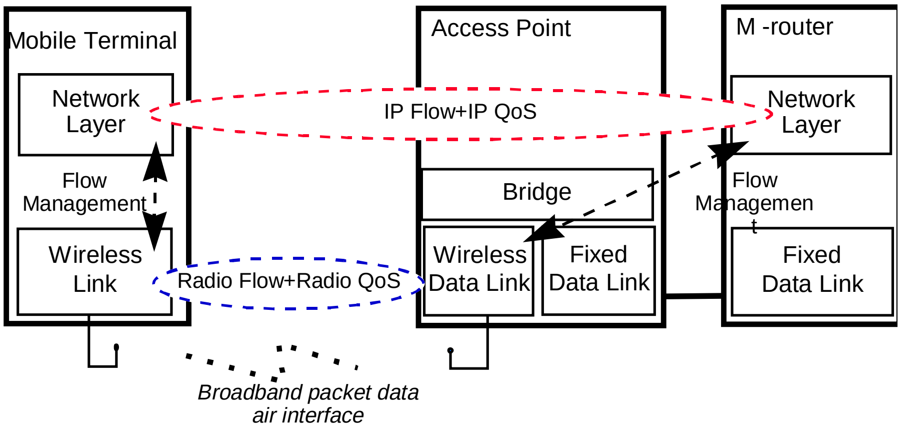


Figure 6: Principal QoS Management Scheme

During terminal registration, a single default connection is created between the terminal and the access network for transmitting IP packets as best-effort data. Once the terminal is registered the system monitors IP traffic flowing between the access network and the backbone and establishes separate connections for traffic with QoS requirements via IP flow detection.

5.2.1 QoS Service Classes

Considerations must also be given to what QoS classes the wireless IP system should support. The current WAND system MAC level QoS classes are closely related to the ATM service categories. In addition to developing approaches for best effort, unicast and multicast delivery, the IETF is also working on real time support within the Integrated Services over Specific Lower Layers (ISSLL) working group. One of the issues being considered is the translation of Internet service class parameters to ATM service class parameters.

Significant progress has been made on determining how to translate Internet service class parameters to ATM service class parameters. [Per95],[Per97] describe how to support best effort applications in UNI 3.1 and UNI 4.0 networks respectively. [Gar97] describes how all three currently defined Internet service classes, (best effort, guaranteed and controlled load), should be mapped to ATM service categories and QoS classes, in both UNI 3.1 and UNI 4.0 networks.

The ATM service categories are as follows:

- Constant Bit Rate (CBR) - Provides delay bounds and no loss by allocating a fixed bandwidth.
- real-time Variable Bit Rate (rt-VBR) - Provides delay and loss bounds for bursty applications.
- non-real-time VBR (nrt-VBR) - Provides loss bounds for bursty applications.
- Available Bit Rate (ABR)] - Attempts to minimise loss and provide a fair share of bandwidth to applications via flow control.
- Unspecified Bit Rate (UBR) - Provides no service guarantees.

The Internet Integrated Services Classes are:

- Guaranteed Service (GS) [She97], guarantees a maximum end-to-end delay, and is intended for audio and video applications with strict delay requirements.
- Controlled Load Service (CL) [Wro97], guarantees to provide a level of service equivalent to best effort service in a lightly loaded network, regardless of network load. This service class is designed for adaptive real-time applications (e.g. applications that can modify their play-out buffer as the end-to-end delay varies).
- Best Effort Service, provides no service guarantees.

Although only three Internet service classes are currently defined, the Internet integrated services protocol suite is sufficiently flexible to allow new service classes to be specified in the future.

The service category suitable for a given VC is highly dependent on the characteristics of the traffic that it will carry. For GS, CBR or rt-VBR should be used depending on the burstiness of the traffic. Both ABR and nrt-VBR are well suited for CL. However, ATM standards currently do not support ABR pt-mpt VCs, which means nrt-VBR should be used for CL multicast applications. Best effort service is best provided by CBR in the core of a network, when a high level of aggregation occurs. If traffic volumes are lighter, UBR or ABR are recommended. Refer to [Per95],[Per97],[Gar97] for more details.

6. MOBILITY MANAGEMENT

Internet users are becoming increasingly mobile. As a result mechanisms are required to allow users to freely move between different IP subnets. Mobility between IP subnets is provided typically via the Mobile IP protocol [Joh98]. In the Internet, traffic is delivered to a terminal on the basis of its IP address. Thus when a mobile terminal moves to a new subnet its IP address must change. Otherwise it is not possible to continue to deliver traffic to the mobile terminal. Mobile IP introduces the concept of a home address and a care-of address.

One of the major objectives of the IP oriented re-design of the WAND system is to provide continuous, native IP services to MTs. The specification of Mobile IP [Joh98],[Per96],[Per97b] suggests that it is well suited to macro mobility management. The system, however must support mobility at both the macro and micro levels. In the context of an IP network, the MT should retain its originally assigned address (home address) so that it is always addressable by other terminals. New addresses (care of addresses, or coa) are assigned to the MT as soon as it enters the area controlled by another M-router. Handovers between APs connected to the same M-router should not result in changes propagating beyond the m-router. The new mobility design aims to implement lossless, possibly soft, handover schemes to protect against high delays or substantial QoS degradation. Finally handovers should be executed rapidly and enable QoS renegotiation. The intra-subnet mobility will be closely aligned with the existing WAND mobility schemes. Inter-subnet mobility will be based on Mobile IP. The remainder of this section describes how Mobile IP operates.

6.1 Mobile IP Overview

In Mobile IPv6 when the MT moves to a new M-router it will gain a new care of address (coa). By default all traffic from correspondent nodes (CNs) (i.e. hosts the MT is currently communicating with) will travel via the Home Agent (HA), and be tunnelled to the M-router. However one of the design goals in IPv6 is to minimise the volume of traffic that has to travel the non-optimised route via the Home Agent. As a result, in addition to telling its HA when it moves, the MT tells all CNs it is communicating with what its new coa is. Each CN can then send traffic directly to the MT in its current location. Similarly the MT sends traffic to the CN with its coa as the source address. The figure below shows an example network with 3 subnets where HA (home agent), a mobile router in a foreign network and a corresponding node in a third subnet communicate. The MT (mobile terminal is about to move from the home subnet to the foreign subnet.

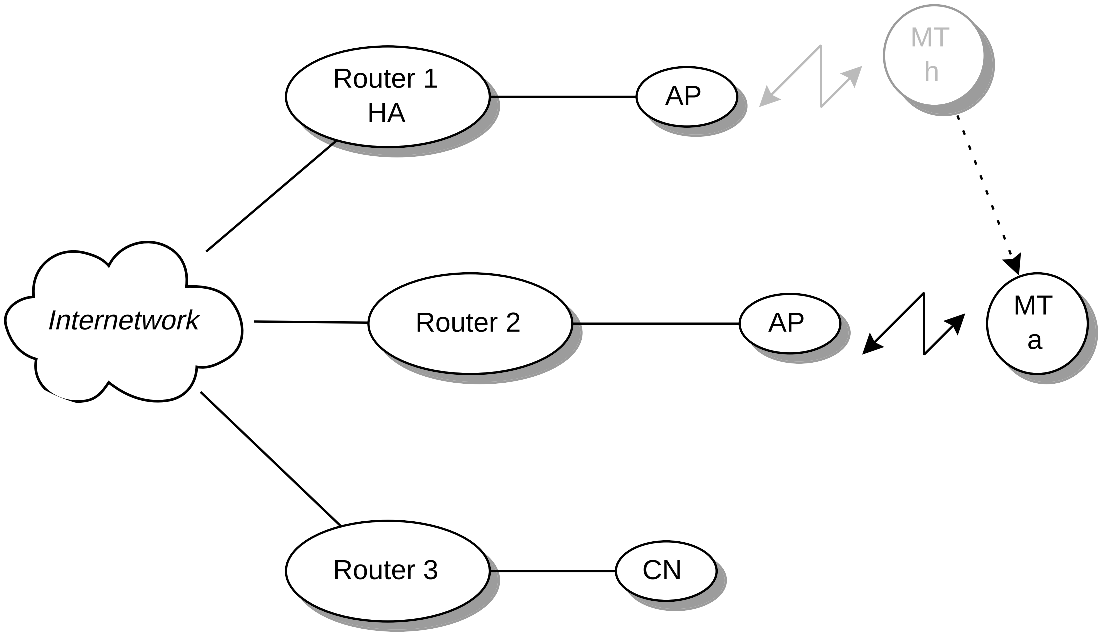


Figure 7: An Example Mobile IP Scenario

The addresses used and communication paths can be derived from the following MSC-like chart:

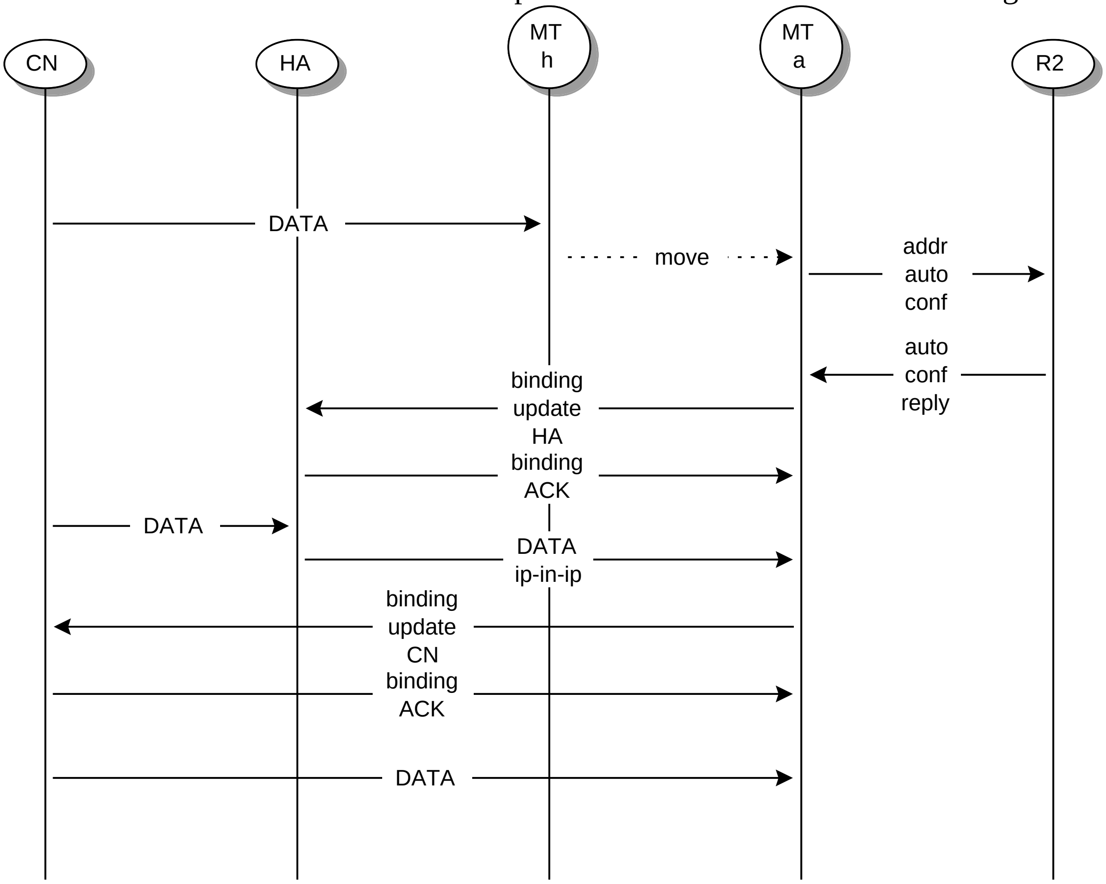


Figure 8: Messages Exchanged when Mobile IP is Employed

Several assumptions were made in this scenario:

- The MT is capable of performing stateless address auto-configuration by combining its MAC address and subnetwork prefix obtained via IPv6 neighbour discovery.
- The CN sends data to the old address before it receives a binding update.
- The HA has already received a binding update (if not, any data sent would be lost).
- Binding updates and ACKS actually reach their destinations.
- After binding to the new location the CN uses direct routing

The MT can also tell its previous M-Router its new coa. The M-router will then behave as a HA for the MT and forward any traffic arriving at its previous destination to the new destination. Note, it is still important that the MT tells its HA its new coa. This will allow traffic from other hosts that are currently not communicating with the MT to reach the MT, via the HA, regardless of its current location.

The address information in packets to and from the MT is as follows:

- CN sends traffic to the MT's coa as the destination address and inserts a routing header containing the MT's home address.
- MT sends traffic with the coa as the source address and inserts a home address option containing the MT's home address.

When traffic arrives at the MT, the routing header is processed which means that the home address becomes the destination address of the packet. Then the IPv6 module processes the packet. Thus all protocols from IPv6 upwards, always see packets arriving to the MTs home address.

When a packet arrives at a CN, the home address must be copied from the home address option to replace the original value of the source header. This means that further processing of the packet (e.g. at the transport layer) is not aware of the coa. Furthermore, because the transport layer at the MT also used the home address the use of the coa is completely transparent to higher layers. However, routers on the path between the CN and MT are aware only of the MT's coa. This is because the router header and home address options are only processed at the end-points of the path.

One of the issues that must be addressed is how to maintain the QoS of the traffic flows to and from the MT when it moves to a new M-router. Rutger's University have developed an approach termed M-RSVP to address this problem [Tal97],[Dat]

6.2 Mobile RSVP (MRSVP)

MRSVP [Tal97], [Dat] has been developed in Rutgers University as an extension to the RSVP protocol. It aims to provide the necessary QoS for mobile nodes as RSVP does for the static ones. To achieve these goals it introduces some new concepts such as proxy reservation agents and some messages to the RSVP protocol.

6.2.1 MRSVP architecture

MRSVP makes the following assumptions:

- The mobile node provides a mobility specification. This is defined as the set of locations the mobile host is expected to visit during the lifetime of the connections.
- The services offered by MRSVP are targeted towards IP networks.
-

6.2.1.1 Service classes

An important concept introduced in MRSVP is the service class distinction. The service model is divided in two main categories. The first one, called Mobility Independent, provides the agreed service in every location the mobile may visit. The second, called Mobility Dependent, provides a high probability, but no guarantee, to provide the agreed service in the locations the mobile node might visit. That means that users requesting this service class may experience severe degradation in the QoS they receive, or even dropping of the connections, even if they remain attached to just one Base Station.

The service classes defined below provide QoS with respect to packet delay bounds, when the mobile host moves within the reach of its mobility specification, and the applications conform to the traffic characteristics agreed.

The service classes defined and the service guarantees for these classes are:

- *Mobility Independent Guaranteed (MIG):* The mobile user receives guaranteed service. Appropriate for intolerant applications that require an absolute packet delay bound.
- *Mobility Independent Predictive (MIP):* The mobile user receives predictive service. Appropriate for more tolerant applications, than the previous class, requiring fairly reliable delay bounds.
- *Mobility Dependent Predictive (MDP):* The mobile user receives predictive service with high probability. Appropriate for tolerant applications that can handle delay variations and disconnections.

6.2.1.2 Admission Control

MRSVP suggests that for users requiring MIG/MIP service, reservations should be made along the routes towards all the possible locations specified in the user’s mobility specification. To improve this inefficient over-reservation scheme, MRSVP allows the reserved but unused resources of the mobility independent class to be used by connections of the mobility dependent class (MDP). Two types of flows are defined to achieve this hybrid scheme.
Active flow: Resources are reserved and data is flowing utilising the reservation.
Passive flow: Resources are reserved, but no data is flowing.

Admission control in MRSVP works as follows. Consider a link of total bandwidth B, A the bandwidth of the current active reservations and P the bandwidth of the passive reservations. If a user requests a mobility independent service across this link, the available bandwidth will be B-A-P. If mobility dependent class is requested, the available bandwidth will be B-A. However, when some or all of the passive reservations becomes active, the mobility dependent flow may suffer QoS degradation.

6.2.2 MRSVP overview

MRSVP introduces the concept of proxy agents to assist in reserving resources along the paths from the sender to the locations specified in the mobility specification. The proxy agent in the mobile node’s current location is called local proxy agent, while the proxy agents in the other locations are called remote proxy agents. The role of the remote proxy agents is to make the passive reservations. The local proxy agent handles the active reservations and is responsible for the routing functionality of its subnet.

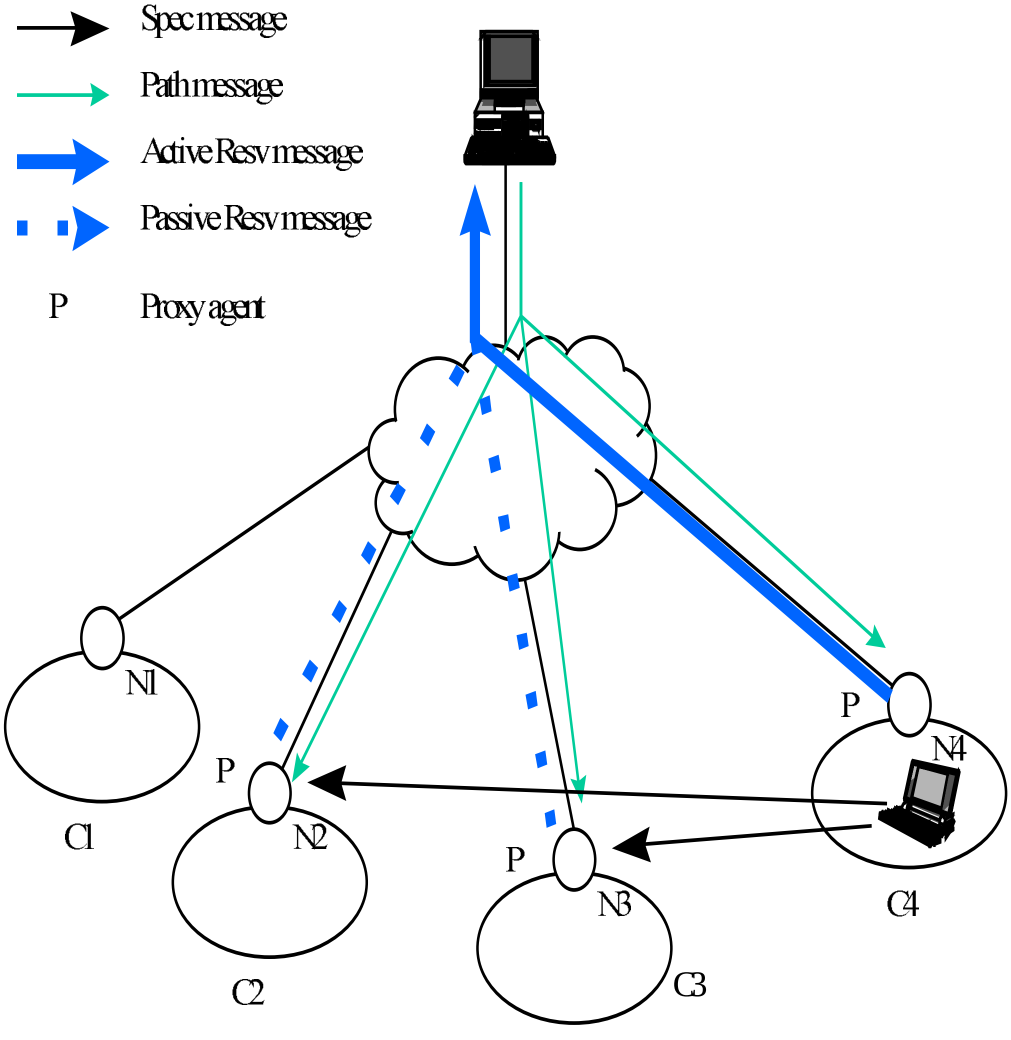


Figure 9: MRSVP messages

The main actions that have to be performed are:

- the set-up of the paths from the sender to the mobile node’s current location and all the locations described in its mobility specification.
- the set-up of active and passive reservations along those paths.

Multicast and unicast flows are supported. The set-up of an active reservation is a straightforward procedure. The local proxy agent handles it with conventional RSVP messages used in the conventional RSVP manner. If a flow is multicast, the remote proxy agents join the multicast group and make the passive reservations along the established paths. If a flow is unicast the paths may be established using unicast or multicast routing.

In Figure 9, the basic operation of the MRSVP protocol is illustrated. The sender is considered to be a node somewhere in the Internet. The mobility specification of the mobile node is C2, C3, C4. Its current location is C4. The spec message is sent to the proxy agents in C2, C3. Nodes N2 and N3, which act as remote proxy agents make passive reservations, while node N4 makes active reservation.

To accommodate the additional functionality of MRSVP, four new messages have been added to the current set of RSVP signals:

1. *Join_group*: The mobile host directs its remote proxy agents to join a multicast group.
2. *Spec*: The mobile host sends the flow specification to its remote proxy agents.
3. *Mspec*: This message contains the mobility specification. It contains a list of the care-of addresses in the foreign subnets and the home address if the home subnet is present in the mobility specification.
4. *Terminate*: The mobile host directs the remote proxy agents to release the reservations.

7. MULTICAST DELIVERY

Multicast is the process of delivering a packet simultaneously to several destinations using a single local transmission [Arm97]. In other words multicast involves more than two users wishing to exchange information [Dio97]. Multicast applications are becoming increasingly popular. Examples of such applications include audio/video conferencing, distributed games, and computer supported collaborative work (CSCW). The interest in multipoint communication spans Internet-style networking, B-ISDN technology including ATM, as well as application and service developers and providers. Hence it is important that future wireless access networks can provide multicast support.

The important issues concerning multicast, or group communication are:

- Group membership management
- Group traffic delivery
- Group addressing scheme

Multicast membership is a dynamic procedure in IP. Thus each host can join or leave a multicast group at any time. A group member can be in another network, so multicast requires the mechanisms for intra-subnet and inter-subnet transmission. Multicast routers are required for the inter-subnet transmission of datagrams.

7.1 Group Membership Management

Group membership management is traditionally performed by the Internet Group Management Protocol (IGMP) [Dee89]. The task of IGMP is to locate hosts belonging to multicast groups in a local subnet. The IGMP entity of a multicast IP router only needs to know the presence of a group in its local network instead of the list of all the group members. Using this information the multicast router decides whether to forward traffic for a given multicast group into that subnet. IGMP has gone through various stages of evolution. The first version of IGMP [Dee89] operates as follows:

In the first version of IGMP [Dee89], the local multicast router periodically invites all local subnet hosts to advertise their group membership, via a query message. On receipt of the query message each host sends a *report* for each group in which it participates after a random delay. A report message for a group is sent to its multicast address so that all the other group members can hear it. This causes the group members to suppress the transmission of their own reports for the specified group. After receiving each report message the multicast router updates its group membership list.

When a host wants to join a group, it sends an unsolicited report message to the multicast address of the group. No explicit action is needed when a host wants to leave a group because the group presence times out if it remains no other member of the group in the local network.

The problem of the above approach is its *leave latency*. Leave latency is the time between the moment the last host leaves a group and when the router is notified that there are no members. The multicast router stops the multicast delivery for a group if it does not receive any report for this group after several queries. During this period of time (leave latency time) redundant multicast packets are delivered to the network.

IGMP v.2 [Fen97] is designed to tune the leave latency. Whenever a host wants to leave a multicast group, it should send an explicit *leave* message to the local multicast router, if it was the last host to reply to a query for that group. A host without sufficient storage to remember whether or not it was the last host to reply may always send a leave message when it leaves a group. The local multicast router just ignores the leave messages for the groups that has group members. The leave group message in IGMP v.2 is only a hint. Thus when the multicast router receives a leave group message, it sends a *group specific query* to determine if there are any remaining members for this specified group. Leave messages are not sufficient to update the group status because the multicast router only maintains a list of active groups in its subnetwork rather than the list of all members for each group. This means that even in IGMP v.2 leave latency is not completely eliminated. Furthermore leave messages that do not actually cause the multicast router to drop a group increase overhead.

Another disadvantage of the normal IGMP operation is that the periodic queries disrupt the operation of every host. This is particularly a problem for battery-powered MTs which must wake up periodically for IGMP processing. However, the number of query messages can be reduced by increasing the query interval. This reduces management overhead but increases leave latency causing inefficiency.

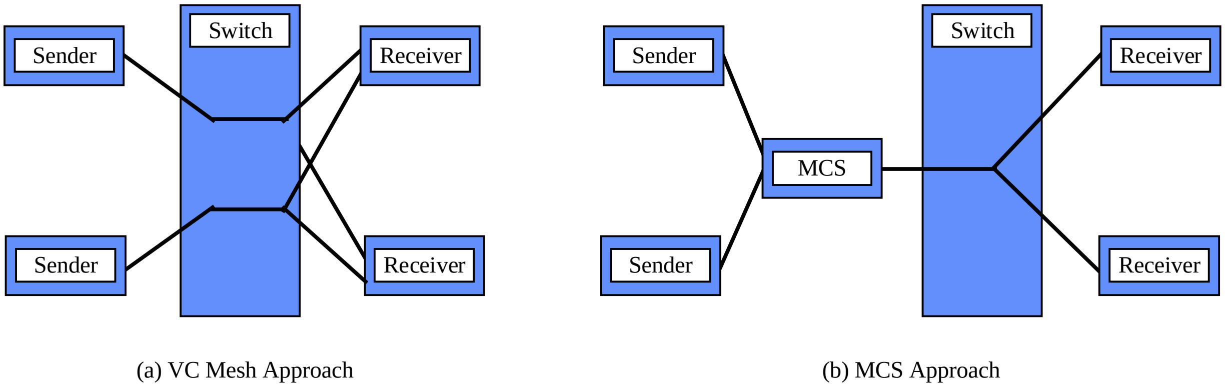
The IGMP standard can be optimised by using explicit join/leave messages as proposed in [Xyl97]. Unlike IGMP v.2, the leave message is not a hint but authoritative information. This enhancement reduces data transmission overhead. However the group management overhead may be either reduced or increased depending on member population (no. of members) and membership dynamics (life time of a group member). If there are lots of members in a given group with a short lifetime, the cost of queries can be ignored. Otherwise it is more efficient that each host sends its own explicit leave group message.

An alternative centralised group membership approach has been proposed for an ATM environment within the IETF. The Multicast Address Resolution Server Approach (MARS) [Arm96] has been designed within the IETF to provide best effort multicast delivery in ATM networks employing UNI 3.1. ATM UNI 3.1 does not provide ATM group addresses, which means that the IP multicast address must be mapped to a list of the ATM addresses of group members. Thus the MARS server must maintain a list of local hosts that are members of each group. Receivers register the multicast groups they wish to join with the MARS. New senders query the MARS to obtain a list of the current multicast group members. As new receivers join the group, the MARS forwards their address details to any active senders. This has a similar impact on group membership management performance to the optimisation proposed in [Xyl97].

7.2 Group Traffic Delivery

Traditionally, the Internet is comprised of shared media based subnets, such as Ethernet and Token Ring. Hence to deliver traffic to multicast group members, the local sender or multicast router simply transmits the multicast traffic onto the shared media, from which all local receivers can obtain the traffic.

In non-broadcast, multiple access subnets, e.g. those based on ATM, an alternative approach is required. The MARS IETF proposed standard describes two approaches illustrated in Figure 10. The VC Mesh approach requires each sender to create a point-to-multipoint (pt-mpt) VC to all receivers. The alternative Multicast Server approach (MCS) requires senders to create pt-pt VCs to a MCS which then creates one pt-mpt VC to all receivers.



(a) VC Mesh Approach

(b) MCS Approach

Figure 10: Comparison of VC Mesh and MCS Multicast Delivery Approaches

The major benefit of the MCS approach is that each sender and receiver only manages one VC, regardless of the number of multicast group members. In contrast the VC Mesh approach requires receivers to maintain a different VC for every sender. It is generally recommended that the MCS approach should be used when VC resources are limited or for dynamic multicast groups, (i.e. where senders and receivers frequently join or leave the group). The VC Mesh approach should be employed when end-to-end delay must be minimised [Fly95],[Bag95],[Arm96].

To deliver a multicast packet in a wireless environment, we need a mechanism to detect the location of MTs that should receive the packet [Ach96]. The following possibilities exist:

- Send a copy of a multicast packet to all APs in the subnet, which they can then broadcast to all attached MTs.
- Send a separate copy of a multicast packet to each MT belonging to the group.
- Send a copy of a multicast packet only to the APs that have at least one member of the group.

The first scheme has the advantage that no location information is needed for a group. The multicast packet is sent to all APs where it can be forwarded to MTs. The disadvantage is the waste of bandwidth since a multicast packet is sent to all APs even if they do not have any members of the group. The second scheme necessitates that the M-router keeps the address of all MTs belonging to a group individually. This approach is not efficient either since it is based on the unicast transmission of packets to each MT. The last approach keeps a list of APs that have MTs that are members of each multicast group. Thus instead of keeping track of each MT individually, the M-router keeps track of the corresponding AP. The reason behind this choice is that the radio is a shared medium and all the multicast packets can be sent without addressing the recipients explicitly. This mechanism only needs to know the APs in which the multicast packet must be transmitted. The other benefit is that if we track the location of each MT individually, every move by each MT will cause a location update while our AP list is updated less frequently

For inter-subnet multicast transmission, a global routing mechanism is needed. A simple solution is that the multicast routers retransmit the multicast packet on all of its interfaces except the one on which the packet is received. More complicated multicast routing protocols have also been developed such as DVMRP (Distance Vector Multicast Routing Protocol) [Wai88], MOSPF (Multicast Open Shortest Path First) [Moy94], PIM (Protocol Independent Multicast) [Dee96] and CBT (Core Based Tree) [Bal93]. Multicast routing protocols operate between routers, hence they are outside the scope of the design of a wireless IP subnet, and not considered further in this document.

7.3 Group Addressing

The group addressing scheme employed within the wireless IP system must also be selected. That is, a decision needs to be made of how to map IP multicast addresses onto MAC level addresses. Group addressing can be handled in the following ways:

- The existing broadcast MAC address together with a pre-reserved MVCI can be used to identify all group communications.
- A pre-reserved MAC address can be used for all group communications. Different groups are identified by different MVCI.
- Each group has different MAC address and MVCI that will be allocated dynamically. No pre-reserved MAC address and MVCI is used in this scheme.

The advantage of the first approach is that no significant modification is necessary in the existing DLC layer since all MTs are capable of receiving data sent over the broadcast channel. Its drawback is that every MT receives all group traffic even if it does not belong to any multicast group. Filtering must be performed in the DLC layer of MTs to discard the packets that are not destined for them.

The second approach limits the reception of group traffic to only those MTs that belong to at least one group. Thus MTs that belong to no group at all will not receive any traffic while a MT which belongs to one group will also receive the traffic of all other groups. Hence there is still a need to filter the traffic in the DLC layer of MTs.

The third approach is most efficient but needs more modifications to the current WAND system than the others. A MAC address is allocated to a group whenever there is a MT in an AP, wishing to join a group and there is no other member of this group in the AP. This address will be used to identify this group as long as there is at least one member of the group in the AP. Since the address allocation is dynamic, the AP must provide a way to inform the MTs in its coverage area about the MAC address associated to this group. The members of the group must memorise this MAC address for their further use. No filtering mechanism is needed in this scheme since only the group members are aware of this MAC address. The third solution is more adapted to a wireless environment where the power consumption is a serious problem for a MT. The MTs receive only the data that are destined to them. Furthermore no software filtering is necessary to discard the unwanted data.

8. APPROACHES FOR CARRYING IP TRAFFIC OVER ATM

Although both the Internet and ATM protocol suites are designed to support all types of application, they do so very differently. The Internet follows a connectionless paradigm, where data is forwarded hop by hop towards the destination. Each Internet Protocol (IP) datagram contains sufficient addressing information to allow each datagram to be routed independently of all others. In contrast, ATM follows a connection oriented paradigm where a connection must be created before data can flow towards the destination. Furthermore, ATM makes routing decisions as the connection is created. This means that ATM cells need to carry a connection identifier only, rather than addressing information. This difference in paradigms means that providing Internet services over ATM is a non-trivial problem that has many possible solutions. Given the current WAND system is ATM based, and the design objective is to provide wireless broadband IP access, with minimal modifications to the existing WAND system, it is valuable to consider alternative approaches for carrying IP traffic over ATM.

8.1 LAN Emulation (LANE)

LAN Emulation LANE [ATM95] and LANEv2 [ATM97a] have been developed by the ATM Forum. LANE allows applications to access an ATM network as if they were running over Ethernet or Token Ring LANs. Hence the primary benefits of LANE are that it allows (1) existing LAN application software to operate unchanged when deployed over ATM networks; and (2) the interconnection of ATM and traditional LANs via bridging technology.

LANE uses a two-phase address resolution protocol. Firstly the IP address is translated into a MAC address via the traditional Internet Address Resolution Protocol (ARP) [Plu82] by emulating a broadcast environment using the Broadcast and Unknown Server (BUS). In the second phase the MAC address is translated to an ATM address by requesting the information from an address resolution server termed the LAN Emulation Server (LES). LANE associates a LAN Emulation Client (LEC) with each ATM host and each bridge that connects legacy LANs to the ATM network. LECs are responsible for resolving ATM addresses and creating any necessary ATM connections. Optionally LANE also provides a LAN Emulation Configuration Server (LECS) to initialise LECs as they join an Emulated LAN (E-LAN). A major benefit of LANE is that it enables multiple E-LANs to be supported simultaneously over a single ATM network, with membership of a given E-LAN independent of physical location.

LANE provides unicast delivery by creating point-to-point (pt-pt) VCs between the sending and receiving LEC. Multicast is provided by forwarding traffic to the BUS, which then broadcasts it to all E-LAN members. LANEv1 provides best effort service only via Unspecified Bit Rate (UBR) VCs. LANEv1 is the only mechanism for IP support in the existing WAND system.

LANE version 2.0 (LANEv2), [ATM97a] was completed in July 1997, and extends LANEv1 by providing real time services and more flexible multicast. LANEv2 allows the BUS to multicast traffic to be forwarded on separate pt-mpt VCs than the general broadcast traffic, to only those LECs that are members of the multicast group. Although LANEv2 is able to provide real time service, it relies on the MAC and the higher layer protocols to provide the QoS request information.

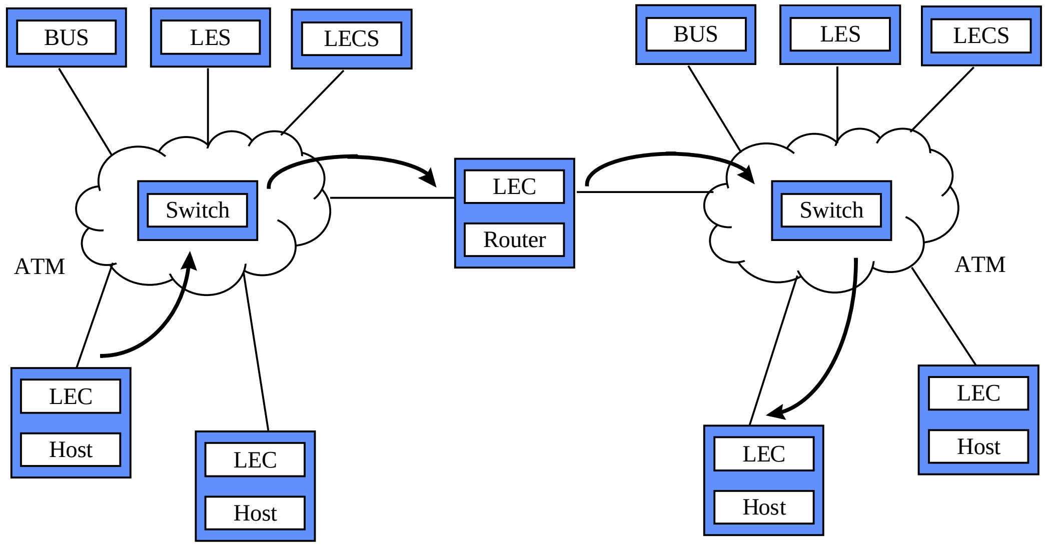


Figure 11: Example LAN Emulation Operation

8.2 Classical IP over ATM

Classical IP over ATM (CLIP) enables unicast, best effort Internet traffic to be transmitted over ATM networks [Lau94],[Hei93],[Per95]. CLIP is currently being extended to support UNI 4.0 ATM networks [Per97], and IPv6 [Arm97]. CLIP follows the traditional Internet hop-by-hop forwarding model where pt-pt VCs are created within ATM subnets, which are connected by standard IP routers (see
Figure 12). The pt-pt VCs can carry best effort traffic using any ATM service category as described in [Per95],[Per97],[Gar97]. Address Resolution is provided via an address resolution server termed ATMARP which maps directly between the IP and ATM address of a host. CLIP is intended to be employed in conjunction with other IETF approaches which provide multicast delivery and real time service support over ATM networks. Such approaches are described later in this document.

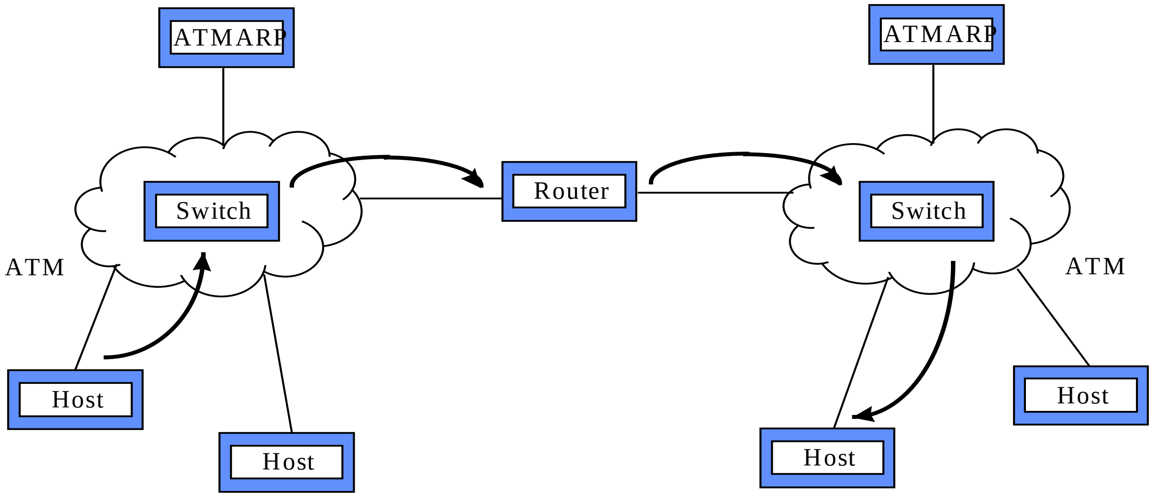


Figure 12: CLIP Example Operation

8.3 Next Hop Resolution Protocol

Many researchers argue that both LANE and CLIP are inefficient when providing Internet services over ATM clouds that are split into multiple subnets. This is because it is possible to create ATM VCs spanning the entire ATM network rather than having to reassemble packets at routers which increases end-to-end delays. The Next Hop Resolution Protocol (NHRP) [Luc97] allows IP addresses to be translated to ATM addresses across subnet boundaries if all neighbouring subnets are ATM. NHRP uses server based address resolution where a series of Next Hop Servers (NHSs) are contacted, one in each subnet, until an NHS is found that stores the requested address information. Note, NHRP is only able to resolve unicast IP addresses.

When either, or both, the sender and receiver are not attached to the ATM cloud, it is still desirable to minimise the number of routers that traffic passes through. NHRP achieves this by creating a direct VC between the ingress and egress routers closest to the senders and receivers [Luc97]. However, when cut-through paths are created across transit ATM networks (i.e. when neither the sender or the receiver reside on the ATM network) persistent routing loops can occur [Col97]. As a result, the current NHRP proposal is limited to the case where the receiver is directly attached to either the ATM cloud, or to an ATM egress router. Rekhter is currently developing extensions to the NHRP proposal to overcome this limitation [Rek97].

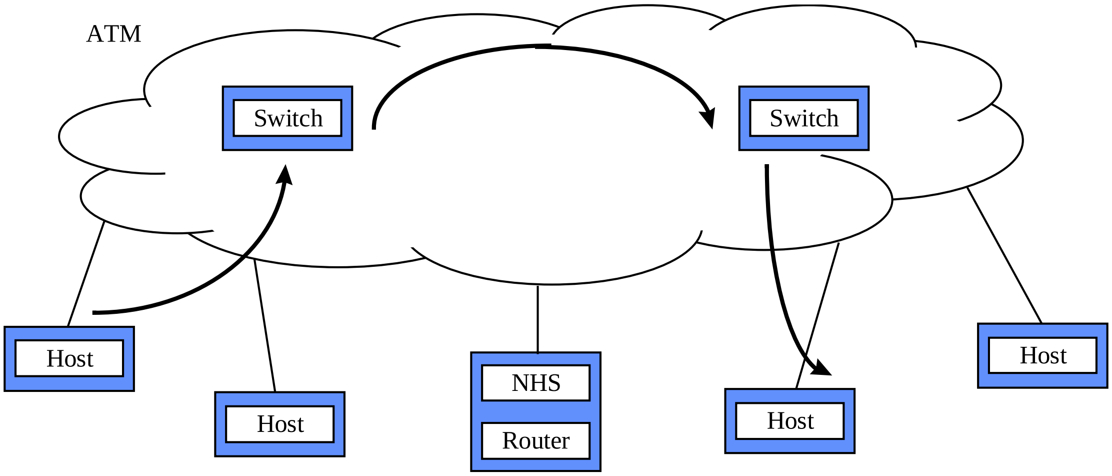


Figure 13: NHRP Example Operation

8.4 Multiprotocol over ATM

Multiprotocol over ATM (MPOA) [ATM97b] integrates LANE and NHRP to provide efficient inter-subnet unicast delivery. Furthermore, like LANEv2, MPOA is designed to support both best effort and real time service classes. MPOA uses the hop-by-hop LANE approach for short lived traffic flows and creates direct VCs using NHRP for flows it identifies as benefiting from removing reassembly delays (e.g. flows containing large volumes of data). MPOA is only able to create direct VCs for unicast flows, multicast flows must be delivered by traditional LANE means and reassembled at subnet boundaries. Sending data via a direct ATM VC in MPOA is more efficient than via the default hop-by-hop LANE path for two reasons. Firstly, it removes router reassembly delays within the MPOA network. Secondly, packets sent via the direct path are no longer encapsulated within MAC frames, reducing the per packet overhead.

MPOA is intended for an environment where some hosts reside on legacy LANs which are connected to the ATM network via edge devices comprising a bridge port, LANE LEC and MPOA Client (MPC). Hosts directly connected to the MPOA network also comprise a LEC and MPC. MPCs use NHRP to create direct VCs when it recognises flows that could benefit from avoiding reassembly at subnet routers. MPCs must perform flow detection on the basis of the destination address, however, other types of flow detection are also allowed.

MPOA also provides MPOA Servers (MPSs) which are co-located with routers. Routers in an MPOA network comprise a LEC, MPS, NHS and standard routing functionality. The MPS interacts with the local NHS and routing function to process MPOA queries from ingress MPCs. Note also, that the MPS can trigger a given MPC to create direct VCs (e.g. to reduce the volume of traffic flowing through the router). Data transfer in a typical MPOA network is depicted below.

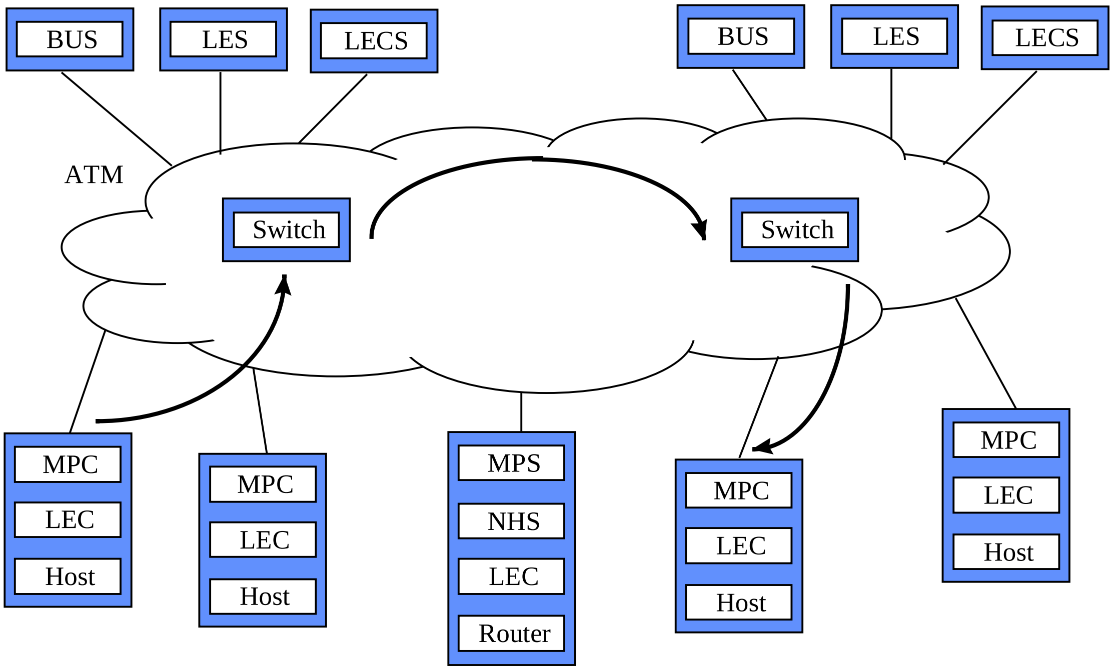


Figure 14: MPOA Operation

8.5 IP Switching

As mentioned above, one of the criticisms of the LANE and CLIP approaches is that they require datagram reassembly at subnet boundaries, even if both subnets are ATM based. NHRP enables direct VCs to be created across multiple ATM subnets by resolving the ATM address of hosts on different subnets. Once the destination or egress router ATM address is determined, standard ATM signalling is used to create the direct VC. An alternative approach is to implement the Internet protocol suite directly above ATM hardware, replacing ATM connection-oriented, hard-state control with Internet connectionless, soft-state control. This has the advantage that lightweight signalling protocols can be employed, reducing the time to create direct VCs (e.g. Ipsilon’s IP switch can create 1000 VCs per second). It also avoids duplication of effort where both Internet and ATM protocols support a given function (e.g. QoS reservation).

Ipsilon Networks (now Nokia Networks) have developed the IP switch which combines ATM hardware and IP control software in this manner [New96a],[New96b],[New96c],[New96d]. An IP switch can either route traffic using standard IP routing protocols or switch a traffic flow at the ATM layer. The IP switch employs a flow classifier to determine which flows should be routed at the IP layer and which should be switched at the ATM layer. [New96b] recommends that long duration, high volume flows, or those requiring QoS be switched and other flows be forwarded at the IP level. Currently IP switches support flows classified on (1) the source and destination IP address, (2) source and destination address and port number, and (3) explicitly reserving resources via RSVP. Hence the IP switch approach follows a data driven approach where direct VCs are created on the basis of the traffic flows seen by each IP switch.

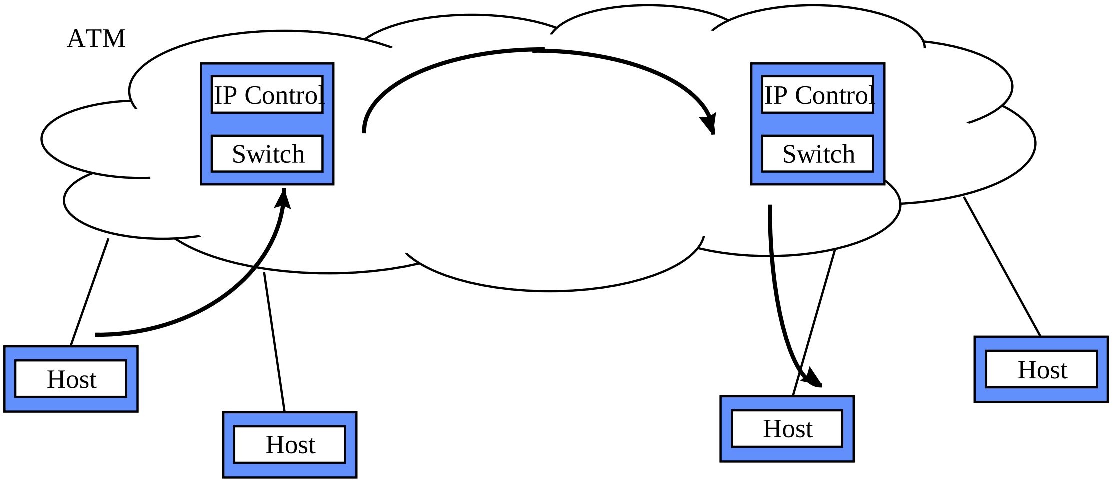


Figure 15: IP Switching Operation

On identifying a flow to be switched, the Ipsilon Flow Management Protocol (IFMP) [New96b], transmits a mapping between the flow and an ATM VCI (and a timer) to the upstream IFMP capable device (another IP switch or IFMP enabled host or router). Hence the direct VC is created on a hop-by-hop basis rather than end-to-end as in standard ATM signalling. Only when a flow has been classified on both the upstream and downstream links can it be switched directly through a given IP switch. Each IFMP capable device can classify flows differently, however within one administrative domain this is unlikely. If all devices employ the same flow classifier, the transmission of the first packet in the flow across the network will create the direct ATM connection.

Assuming all IP switches in a network follow the same flow classification policy, redirected multicast flows will follow the VC Mesh delivery mechanism described above where a pt-mpt VC will be created from each sender to all receivers. The standard Internet Group Management Protocol (IGMP) can be used to identify the receivers in each multicast group. QoS support will make use of the queuing and scheduling capabilities of the underlying ATM switch and use RSVP to signal resource requirements. Policing can be provided by configuring the UPC hardware in the ATM switch according to the RSVP flowspec.

Another interesting feature of the IP switch approach is that it reduces transmission delays for switched flows. This is achieved by removing the LLC/SNAP header and any IP header fields used to identify the flow, since these fields will be consistent for all packets carried on that flow. The header can then be reconstructed at the edge of the ATM switched flow using stored headers.

In terms of implementation, an IP switch comprises any ATM switch that supports the Generic Switch Management Protocol (GSMP) [New96d] and an IP switch controller which is able to operate on a high end Pentium Pro. GSMP has been implemented on 8 different ATM switches and involves around 2000 lines of code. A reference implementation is available [Ips97] and Ipsilon has found it typically takes 1-2 weeks to implement GSMP on a new switch design.

One restriction of the Ipsilon approach is that all ATM switches must be replaced by IP switches. A similar approach termed Cell Switched Routing, proposed by Toshiba, does not suffer from this restriction. Instead it enables standard ATM subnets to be connected by Cell Switch Routers (CSRs) [Kat97],[Kat97b]. The CSR approach is intended only for applications that require QoS guarantees since it relies on RSVP to provide the mapping between the flow and VCI. Goto has proposed a session identification protocol SINP that overcomes this restriction by allowing an upstream node to announce its intention to send a flow on a given VC and then start sending [Got95].

8.6 Multiprotocol Label Switching

The MPLS effort was initiated to develop a generic label switching approach combining the concepts from approaches proposed by Cisco (Tag Switching), IBM (ARIS), Toshiba (CSR), Ipsilon and others. The underlying objective of MPLS is to simplify forwarding by using a short fixed length label to identify a traffic stream, rather than processing several header fields. Hence MPLS is being designed to apply to any link layer network.

Multiprotocol Label Switching [Cal97],[Ros98] provides a non destination based forwarding mechanism which can be used to forward IP packets. Instead of looking up a routing table using a longest prefix match criterion, a Label Switch Router (LSR) performs a perfect match on a label that is associated with a packet in an MPLS domain. The label identifies an entry in a table in the LSR forwarding engine. This entry contains the next hop, the outgoing interface and a possible new label to be associated with the packet. The concatenation of these associations from the ingress to an egress of an MPLS domain creates a Label Switched Path (LSP). The associations can be data driven (like the IP switching approach) or control driven. Control driven label assignment is the one that is focused upon by the MPLS WG. Based on a Label Distribution Protocol (LDP), IP routing protocols and some signalling to set-up explicit paths (which can be used for traffic engineering) an LSR establishes associations between an incoming label value, a FEC (Forwarding Equivalence Class), and the outgoing label value that has to be used. A FEC identifies which packets should be forwarded into an MPLS domain with a given label value. A FEC can be identified at a MPLS boundary based on MF (Multiple Fields) classification or can be traffic belonging to a single customer, arriving from a certain interface, which has to be handled in a proper way to provide VPN services. Edge LSRs must perform FEC identification and use the proper label to forward packets. Core LSR will have to maintain

LSP integrity by keeping a LIB (Label Information Base) up to date with respect to routing or explicit routing changes, and forward packets based on label value look-up only. The label may be a field in a L2 frame or cell (e.g. the DLCI, the VPI…), a shim between L2 and L3, or a field in L3 packets (e.g. the flow label in IPv6). Figure 16 shows a typical MPLS domain.

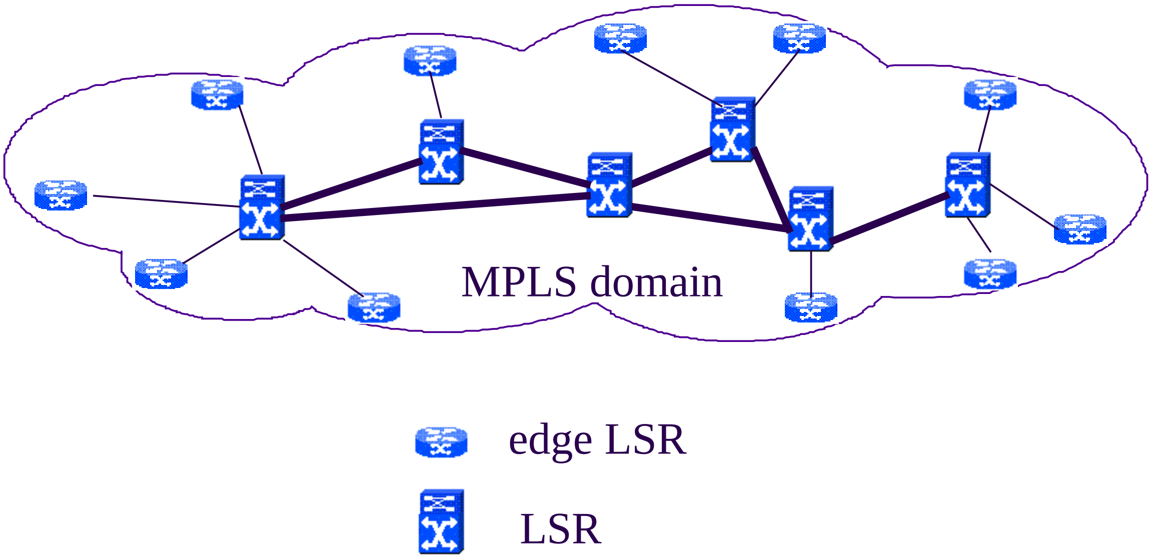
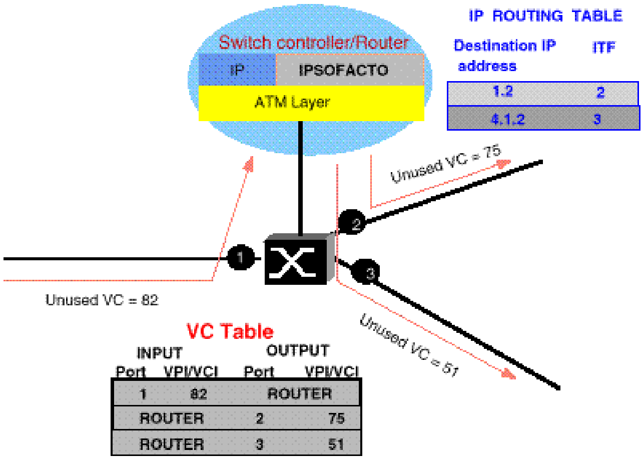


Figure 16: An Example MPLS Domain

8.7 IPSOFACTO

Ipsofacto [Ach97], [Ach98] has been proposed by the C&C Research Labs of NEC, USA. Similarly to IP Switching, Ipsofacto completely eliminates conventional ATM signalling. The basic idea behind the operation of Ipsofacto is that all unused VCs on an input port of a switch are mapped to the switch control processor (see Figure 17). A cell level switched path for data forwarding is established in the following way: a sender selects an unused VC on an outgoing link to forward the first packet of a new flow. This is received by the switch processor at the downstream end of the link, which then selects an outgoing link, based on its IP routing tables, and an unused VC within the selected link. The first packet of the flow is then forwarded through this unused VC on the selected link. Subsequently, the switch processor creates a ATM level cross-connection for the mapping (input port, input VC) → (output port, output VC). Subsequent packets are then switched at the ATM level (Figure 18) thus, eliminating the need for IP datagram forwarding through the switch control processor.



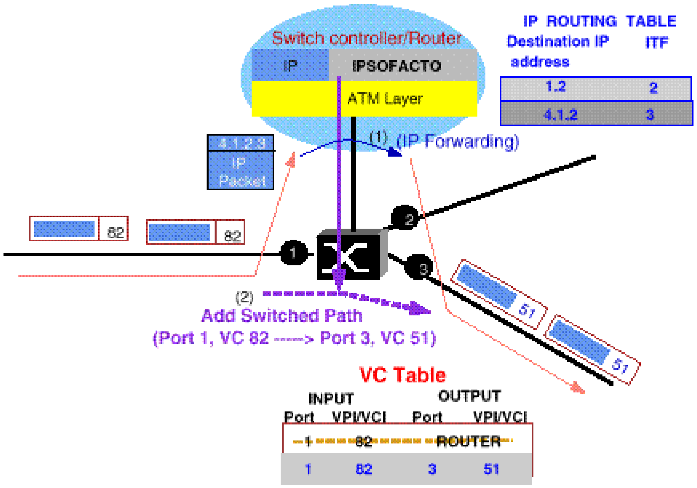
IP ROUTING TABLE

Destination IP address	ITF
1.2	2
4.1.2	3

VC Table

INPUT		OUTPUT	
Port	VPI/VCI	Port	VPI/VCI
1	82	ROUTER	
ROUTER		2	75
ROUTER		3	51

Figure 17: Traffic from unused VCs is always diverted to the Switch Controller



IP ROUTING TABLE

Destination IP address	ITF
1.2	2
4.1.2	3

VC Table

INPUT		OUTPUT	
Port	VPI/VCI	Port	VPI/VCI
~~1~~	~~82~~	~~ROUTER~~	
1	82	3	51

Figure 18: A Cross-connection is installed in the ATM switch

IP control messages (ICMP packets, SYN/FIN TCP packets, etc.) are always exchanged over a predefined “control VC” which is terminated at the switch control processor (i.e., are subject to the routing process and never transmitted over switched paths). Since control messages are always routed (i.e., forwarded through the control processor), forwarding state information for each flow can be maintained at the switch control processor. When such state information is removed from the controller the corresponding switched path is released and the associated VCs are considered unused.

When an outgoing VC, that is currently in use, is reclaimed back, this action is preceded by marking the VC to be in a transient state, sending a RECLAIM message downstream and waiting a RECLAIM-ACK message in reply. When the downstream node receives a RECLAIM message, it marks the corresponding VC as unused, advances the RECLAIM message further downstream and responds with a RECLAIM-ACK signal to the upstream node.

Unlike other schemes like Ipsilon's IP Switching or Toshiba's CSR, Ipsofacto informs the downstream node on the selected VC implicitly through its use. In contrast Ipsilon uses IFMP for signalling upstream nodes on the selection of a VC; the selection of the VC is the responsibility of the downstream node. CSR adopts the reverse strategy as it involves the upstream node to be able to select VCs (notification is sent downstream through the FANP).

TCP communication is preceded by the three-way handshake involving SYN and SYN-ACK messages. With Ipsofacto the switched path is set-up when the first SYN packet is forwarded; thus an end-to-end switched path is available prior to actual data transfer. Similarly to this approach, the tear down of the TCP connection (the marking of the associated VCs are unused) can be forced by sending the FIN and FIN-ACK messages through the switch controller instead of the switched path (i.e., by handling the relevant packets as control packets).

8.8 IBM QoS Capable Switch-Router

The IBM QoS-capable Switch-Router (QSR) [Bas97], consists of an ATM switch fabric to which a number of intelligent adapters are attached, with each adapter also supporting an interface to an external OC-3 link. The adapters (also termed Forwarding Engines, FE) are capable of routing and switching packets, and include hardware support for providing different levels of quality-of-service. The switch and the intelligent adapters are controlled by a Control Engine (CE) that resides in a separate adapter and is responsible for running the different routing protocols that QSR implements, and for handling all aspects of resources management, including signalling which is supported through the RSVP protocol. The QSR adapters can be distinguished into port and trunk adapters. Trunks interconnect adjacent QSRs and are optimised for speed and performance. Port adapters provide flexible access functions, such as the packet classification functions needed at the periphery of the network. These classifier functions, which precede the regular IP forwarding loop, identify packets that are to be granted special service and place the matched packets onto particular layer-2 connections. From there on, i.e., on trunk interfaces, these packets are handled at layer-2 (switched) with appropriate scheduling support provided by the hardware. The classification function is based on source and destination addresses and transport level port numbers embedded in the packet.

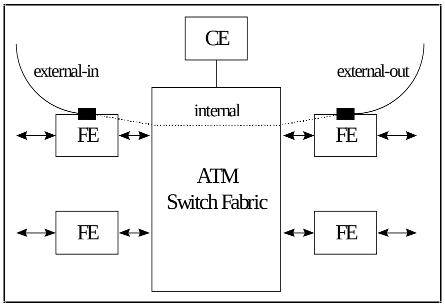


Figure 19: QoS capable Switch-Router

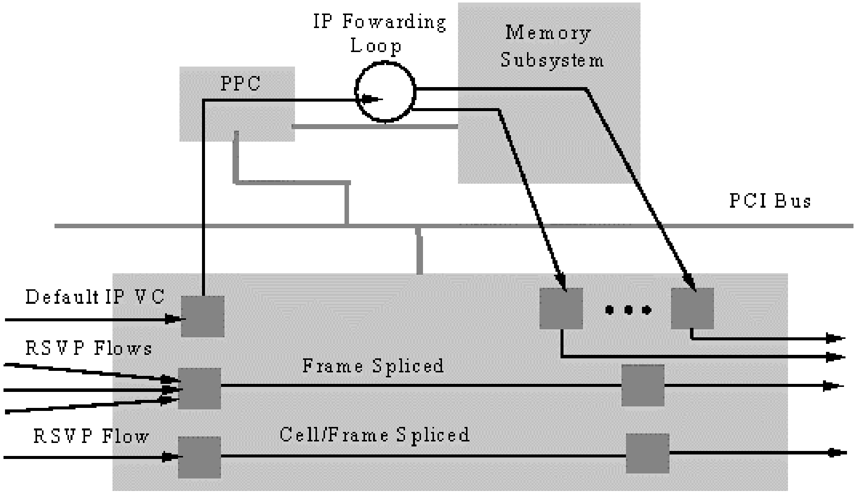


Figure 20: Data Paths through the Forwarding Engine

The general structure of data flows through a QSR is shown in Figure 19. The data flow comprises three basic VC segments: external-in, internal, and external-out. Connections between the segments can be made either at layer 3 (routed VCs) or directly at layer 2 (switched VCs). Figure 20 shows these different connectivity options for data flows through an FE. Connectivity selection is performed for each VC, and all cells/packets arriving on a given VC are either routed or switched. The system supports both store-and-forward and cut-through forwarding options. In the cut-through mode, which is to be used to provide service differentiation, both cell-based and frame-based forwarding is needed. The cell-based approach minimises latency and storage requirements and is preferred for unicast flows while the frame-based forwarding is needed to allow merging of flows.

Cells arriving from the link (or the switch) on “routed” VCs are buffered and reassembled into packets in the packet memory of the Rx (or Tx) subsystem of a FE. The headers of the reassembled IP packets are then processed (a next hop look-up is performed) and modified. The modified headers are moved back to the packet memory and the associated packet is then readied for transmission to the switch (link) on the corresponding VC. Cells arriving from the link (switch) on “switched” VCs can be handled in either one of two ways. They can be buffered and reassembled into packets in the packet memory of the Rx module of the FE (or the Tx module) before being readied for transmission to the switch (link). Alternatively, cells can be made available for transmission immediately after they have been received. These modes correspond to the frame and cell cut-through forwarding modes mentioned earlier, and can be configured for each VC.

The CE implements the RSVP protocol, with some extensions needed to support the mapping of RSVP flows onto switched connections. This essentially requires the ability to communicate the identity of the VC that is to be used for a given RSVP flow. In the current QSR implementation, this information is carried in the PATH messages (i.e., the sender selects the VC to be used) as they travel from QSR to QSR. In particular, VCI information is piggybacked into the LIH field of PATH messages. Reservations are activated upon receipt of a RESV message, and serve as the trigger to the forwarding of data packets onto the switched path.

After extracting the VCI information from the PATH message, the RSVP protocol contacts the resource management entity found in the CE. It first notifies it of the VCI to be used by the new flow on the link terminating at the input adapter. Resource management verifies that this VCI is not already in use. Assuming it is not, RSVP then provides resource management with the identity of the output adapter/link on which the PATH message is to be forwarded (RSVP obtains such information from routing). Resource management then returns the VCI values of both the internal VC that will be used through the switch between the input and output, and of the VC to be used on the specified output link. The latter is inserted into the LIH field of the Path message that is sent to the next downstream node. The above applies to unicast flows, but multicast flows are treated similarly, simply by indicating the multicast nature of the flow through a special flag. The use of this flag signals to resource management that the internal VC needs to be allocated from the pool of broadcast VCs. VCs for each of the links corresponding to outputs associated with the multicast flow are allocated one at the time, through repeated AddParty() calls that each time specify the identity of a new output.

Specifically, upon receiving a RESV message specifying a given service class and service parameters, the RSVP protocol communicates this information to resource management. Resource management then performs call admission and resources allocation, and assuming that this step is successful, it then triggers “splicing” of the different VC segments associated with the flow. Specifically, the internal VC is spliced in the output adapter onto the VC assigned to the flow on the output link. Similarly, on the input, the VC assigned to the flow on the incoming link is spliced onto the internal VC. This splicing ensures that from that point on, packets are forwarded directly at layer 2 through the QSR. However, there is still the need to identify data packets belonging to RSVP flows as they enter the first QSR box on their path.

This amounts to updating the classifier on the ingress port adapter of the first QSR box. Triggering this update is again under the responsibility of the RSVP protocol in the CE, and is performed upon receipt of the first RESV message. It is performed by sending a control message to a client stub residing in the ingress port adapter. This message specifies the necessary information (source and destination addresses and port numbers) to identify the corresponding packets. The client stub then updates the data path classifier accordingly, so that packets matching this criteria get immediately forwarded onto the internal VC associated with the flow.

RESV messages do not carry the assigned VCI value back in the LIH field. This is because a single RESV message may be carrying multiple reservations, and including all the corresponding VCI values cannot be done through a single LIH field. In addition, this would unnecessarily overload the RESV messages. Furthermore, returning those values has little benefit since the receipt of a RESV message implicitly acknowledges receipt of the associated Path messages and, therefore, of the VCI value it carried.

Reservation and flow removals (ResvTear and PathTear) are handled in a symmetric fashion. Upon receipt of a ResvTear message, the associated VCs are “unspliced” and packet forwarding returns to the default IP data path. In the case of a multicast flow, one needs to determine if the reservation was the last active one, at which point unsplicing of the internal VC at the input is also done. De-allocation of the VCs is only performed upon receipt of a PathTear (or the time-out of the associated path state), at which point both the internal and external outgoing VCs are returned to the pool of free VCs.

Even if the data packets of an RSVP flow are eventually carried over a switched connection, this does not apply to control messages such as PATH and RESV messages. These continue to follow the “hop-by-hop” routed path. This is worth emphasising as it implies that the path followed by the switched connections remains entirely under the control of the IP routing protocols. For example, when the RSVP protocol detects a route change, the switched path is immediately taken down from that point on, and will be re-established along the new path as PATH and RESV messages get received.

RSVP data packets arriving at an ingress port adapter are intercepted by the classifier. The classifier function is needed since the identity of the incoming VC is typically not sufficient to identify the data packets as belonging to an individual flow (multiple flows as well as non-RSVP packets can currently be multiplexed on any incoming port VC). The classifier uses a single hash-based look-up that combines the source address and port number, destination address and port number, and protocol type into a hash key. Collisions in the hash table are handled using simple chaining. The classifier identifies packets belonging to an established (a reservation is in place) RSVP flow and forwards them directly onto the corresponding VC. In the output adapters, for both unicast and multicast flows, the VC on which data RSVP packets arrive is directly spliced onto a point-to-point VC going out on the link. At the input trunk adapter of the next QSR on the path, this VC is again directly spliced onto the VC (point-to-point or broadcast, depending on the type of the flow) that takes it across the switch of this next QSR. This process repeats until the egress port adapter is reached. Hence, the packets are processed only at the switched level until they reach the egress port adapter.

The QSR also supports the “provisioned IP” service for reserving resources, with the aid of RSVP, on behalf of aggregate traffic streams. The “provisioned IP” service is constructed using a special packet classifier function at an ingress port and a switched IP tunnel connecting the ingress port to the appropriate egress port. In the QSR system, the establishment of switched pipes and the associated updates of the classifier function on port adapters are implemented through an application, Provisioned Switched IP (PSIP), that resides in the CE. The data path of a “default” IP packet starts with its entrance into the QSR network at a port adapter. Connectivity to this port adapter is provided through an ATM VC being set-up using CLIP (Classical IP over ATM) for unicast traffic, or the MARS (Multicast Address Resolution Server) protocol for multicast traffic. Packets arriving at a port adapter are reassembled by the Rx module of the FE. Incoming VCs may either be PVCs or SVCs. Since SVC functionality is provided , port adapters implement the ATM UNI signalling stack.

QSR supports both the Controlled Load and Guaranteed Service Integrated Services models.

Section II: Proposed Wireless IP Architecture

9. OVERVIEW

Previous chapters have covered the most important design areas proposing several relevant solutions e.g. for the system architecture, mobility management, QoS management etc. This section combines the results together and describes the complete system concept.

10. SYSTEM ARCHITECTURE

10.1 General Overview

The M-router has full TCP/IP protocol functionality. It performs standard IP routing forwarding packets to the RAN interface and embeds wireless specific control functions. The M-router classifies incoming IP packet flows and relays them via the corresponding access point to the mobile terminal using suitable QoS characteristics. The wireless extension controls radio flows, terminal mobility and location management. The M-router controls the access points using a specific control protocol. The access point implements a LAN bridge that multiplexes the IP flows into radio flows.

The mobile terminal includes all standard TCP/IP protocols and wireless specific control services. The control messages are transparently sent between the M-router and terminals utilising control functions. Figure 21 illustrates the functional architecture.

The M-router segments and re-assembles IP packets into segments that fit into radio link packets. The Segmentation And Re-assembly (SAR) blocks handle the segmentation of packets between the mobile terminal and the M-router. The access point only transparently relays the segmented packets between the radio access network and the fixed network. The current BRAN radio utilises ATM like segmentation (AAL), which segments the IP packets into 48-octet cells.

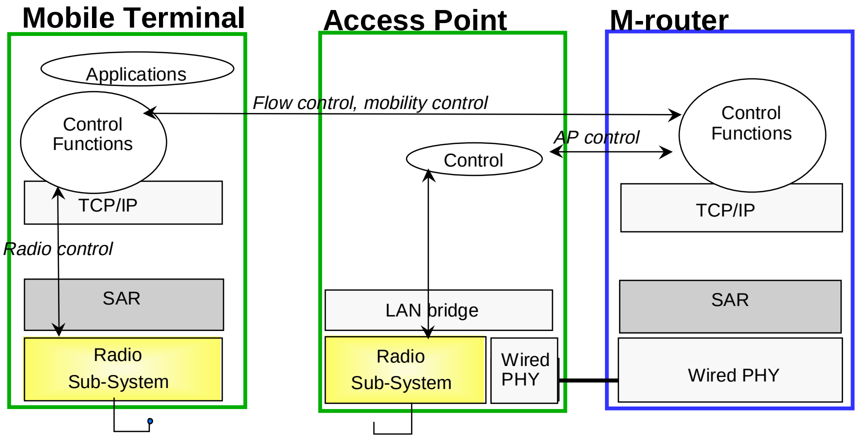


Figure 21: Overview of the Functional Architecture

10.2 Transmission Link Alternatives

The system offers two physical transmission technologies for connecting access points to the M-router: ATM and Ethernet. Here ATM refers to the transport link without control signalling. In this case ATM provides only segmentation & re-assembly and transmission services for IP traffic. The ATM option is intended for public telecommunication networks owned by a public operator while the Ethernet option offers an ideal solution for private wireless business LANs, such as customer-premises networks. It is assumed that the public hot spots are built as separate systems, which offers a good opportunity to deploy ATM links for connecting M-routers to the backbone networks. Figure 22 illustrates the transmission link options.

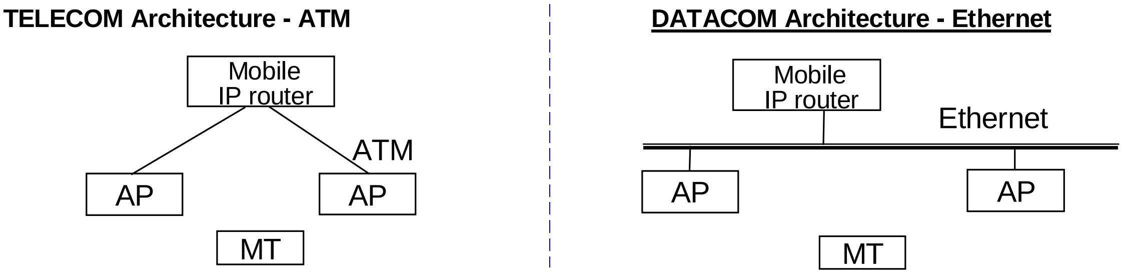


Figure 22: Transmission Link Alternatives

In the Ethernet case all access points belong to the same Ethernet segment which is connected to the router e.g. through a hub. The access points in the same Ethernet segment have the same IP sub-network address. Therefore, they comprise one IP sub-network (also termed a mobility domain).

10.2.1 Packet Encapsulation in the ATM Case

In the ATM link the M-router port identifies the access point uniquely. The IP packets are encapsulated into AAL5 frames between the M-router and the mobile terminals. The AAL5 layer segments the packets into 48-octet long ATM cell payloads. In this case the access point passes the ATM cells transparently to the M-router.

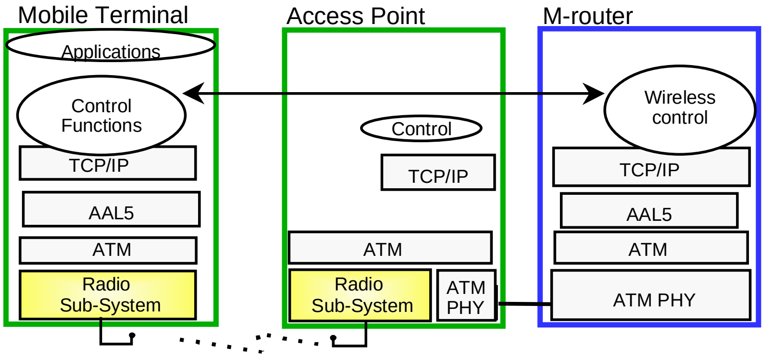


Figure 23: Data Plane in ATM Case

Figure 24 illustrates the packet encapsulation in the ATM case.

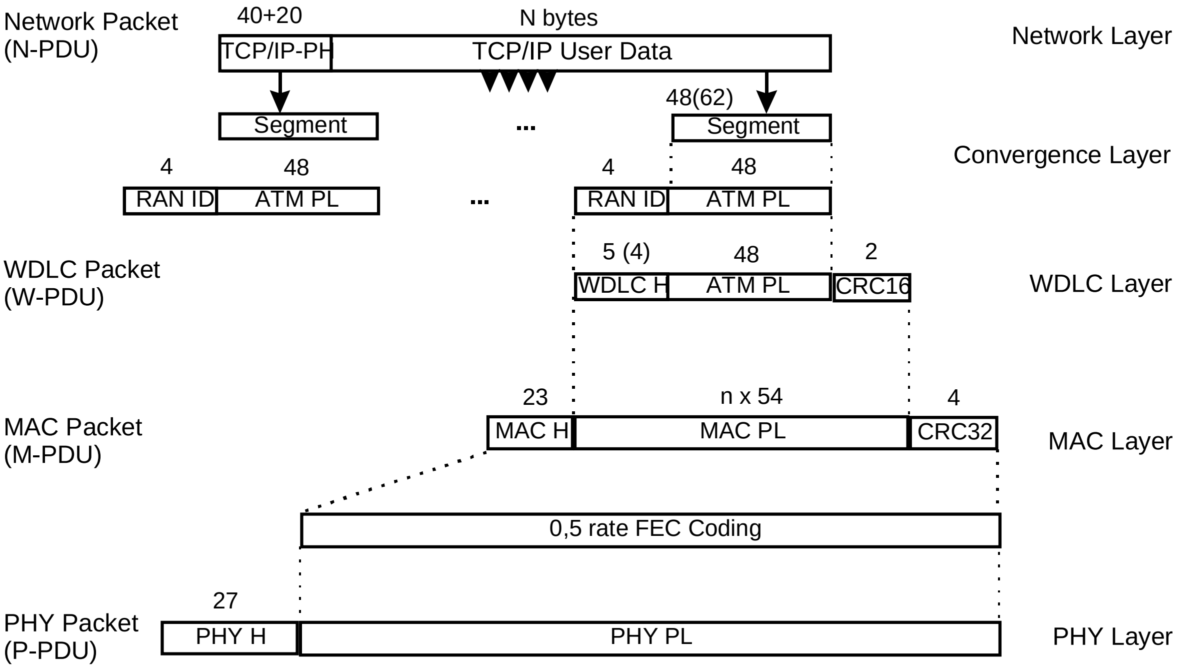


Figure 24: Packet Encapsulation if ATM transmission links are used

10.2.2 Packet Encapsulation in the Ethernet Case

In the Ethernet link IEEE MAC addresses are used to identify access points and the M-router ports. The M-router and mobile terminals encapsulate IP packets inside Ethernet frames. Before packets are passed into the Ethernet layer (SAR) they are passed through a proprietary IP/Ethernet convergence layer. The convergence layer adds a dedicated flow label between the IP packet and the Ethernet header. This flow label corresponds to ATM VPI/VCI values. It is decoded in the access point (bridge) which multiplexes Ethernet packets into the correct radio flows. The mapping between radio flows and Ethernet frames is done in the convergence layer.

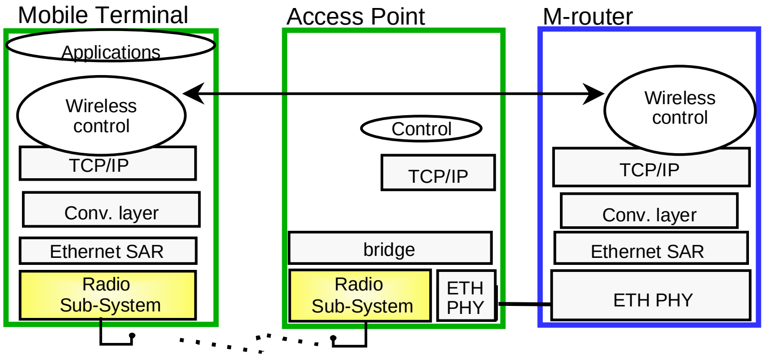


Figure 25: Data Plane in the Ethernet Case

In both cases the convergence layer marks the IP packets with a radio access network specific RAN identifier. In the case of ATM the RAN_ID corresponds to the VPI/VCI value while the Ethernet uses a random 24-bit identifier allocated by the M-router. The first 8-bits of the RAN_ID are used to identify the connection and the remaining 16-bits for identifying the terminal. The RAN_ID is unique within the access point. The ATM case uses the M-router port to detect the access point while the Ethernet case utilises AP IEEE MAC addresses. If ATM transmission links are used the flow mapping and the use of the RAN_ID (VPI/VCI) is trivial.

In the Ethernet case the RAN_ID can be any random 24-bit identifier which is added between the IP packet and Ethernet headers. However, to achieve a scaleable system specification the allocation of 24 bits in Ethernet case was selected so that it corresponds to the ATM case. The first 8 bits identity the connection (radio flow) and the next 16 bits identify the terminal. Consequently, the encapsulation of Ethernet packets is compatible with the ATM case, except that the segment size differs (ATM has 48-bits). Note: A dedicated protocol ID has to be defined and added in the Ethernet header for indicating the existence of RAN_ID in front of IP packet. Figure 26 illustrates the resulting Ethernet packet.

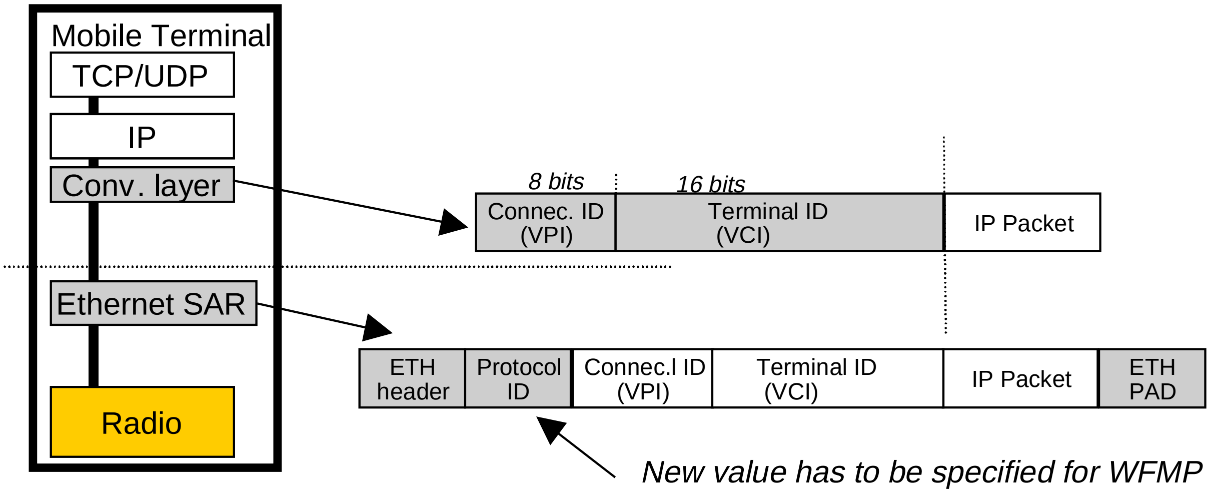


Figure 26: Flow Mapping in the Ethernet Transmission Link Case

The only significant difference between these architectures is the transmission link and the interface between the radio sub-system and the transmission link. Otherwise both solutions are identical. Therefore, it can be stated that we have defined *a single system architecture that offers two optional transmission mechanisms: ATM and Ethernet.* The remainder of this document will consider both alternatives in parallel.

11. FUNCTIONAL ARCHITECTURE

The detailed system architecture is illustrated in the figure below. The main external interfaces are listed in the following tables.

Table 2: External Control Interfaces

IF#	Interface	Explanation
1.	MMC – MMC	Mobility management messages between terminal and M-router. Mobility management messaging is used as a new terminal registers in the network and in the case of handovers.
2.	WFMP – WFMP	Flow management control signalling messages. This is used for establishing and releasing radio flows.
3.	MCP – MCP	Mobile Control Protocol (MCP) provides a reliable peer-to-peer protocol for transmitting WFMP and MMC messages between the mobile terminal and the M-router. MCP is used for all wireless specific signalling.
4.	Radio control messages	Radio control messages are used for transmitting radio link control messages. For instance terminal association and radio flow (MVC) control signalling is carried here.
5.	APCP interface	Access Point Control Protocol (APCP) is used for sending radio link control and radio resource management messages between the access points and the mobile router

The external control interfaces define the logical interface between the mobile terminal and the radio access network (access points, M-router) and between the radio access network and the core network (AP-M-router). The external interfaces have to be standardised, if the target is to define compatible standard systems that can be composed of devices from different manufacturers.

In addition to the standard interfaces the system also has several important internal control interfaces, which are listed in the table below:

Table 3: Internal Control Interfaces

IF#	Interface	Explanation
6.	Wireless QoS controller – WFMP interface	This is an internal interface which is used for transmitting flow establishment requests and QoS information between the QoS manager and the WFMP. As MR-WFMP detects a new flow it queries the radio link priority from the QoS manager. In a real implementation QoS manager and WFMP can be integrated into a single entity which removes interface 6.

7.	Wireless QoS controller – RSVP interface	Wireless specific QoS controller interacts with RSVP module for obtaining resource reservations and converting those into radio resource reservations and radio QoS. The RSVP module requests resources from the wireless QoS manager via this interface.
8.	Wireless QoS controller – MMC interface	This interface is used in the case of handovers. The MMC module informs the QoS controller about the handover and requests QoS controller to re-establish wireless flows. This establishment request is then forwarded to WFMP entity.

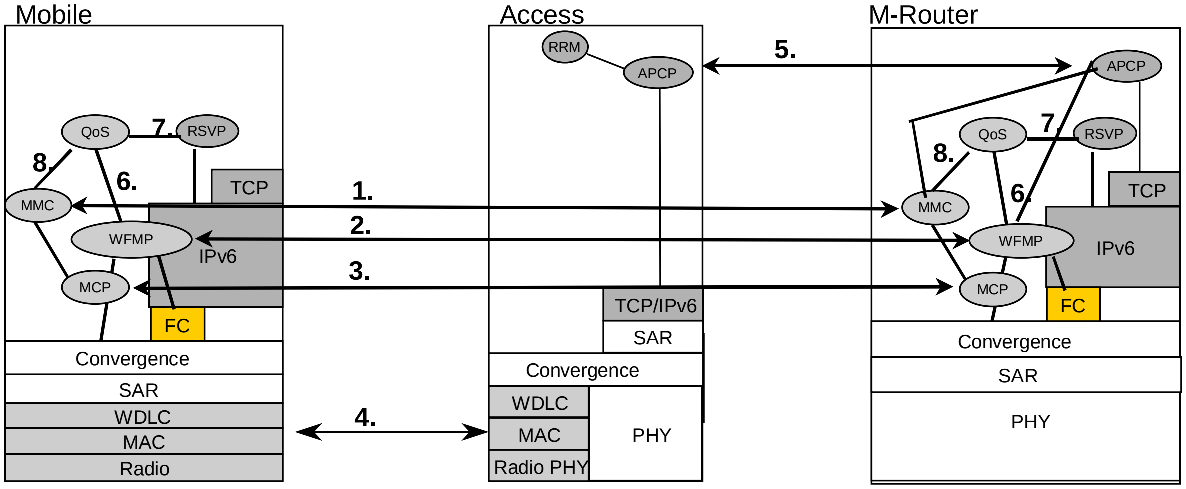


Figure 27: System Architecture and the Main Interfaces

The system architecture includes the following WAND specific functional blocks:

QoS - Wireless QoS controller: This entity allocates the radio link QoS for the IP packets. The QoS controller has an interface to the RSVP module which can give explicit QoS requirements, such as delay and bandwidth, for the IP flow. If no explicit QoS parameters are available the QoS manager assigns the QoS on the basis of the DS field (differentiated services) or on port information (standard applications). As WFMP detects a flow it informs the MR-QoS entity of the evaluated packet throughput, this information can also be employed to allocate radio link QoS. The QoS controller transmits the allocated radio link QoS values to the WFMP entity that then establishes radio flows with the selected QoS.

Mobility Management Controller (MMC): The MMC entity is responsible for terminal mobility management. The MR-MMC has a database that contains information about the registered terminals and their current location (access point). During terminal registration MMC can be used to authenticate the user. The mobile terminal MMC initialises the handover by sending a handover_request message to the MR-MMC which checks the radio resources in the new access point and requests WFMP to establish new radio flows in the new access point and to release the old radio flows.

Wireless Flow Management Protocol (WFMP): WFMP entity manages the radio flows. It detects IP traffic and classifies IP flows. As WFMP detects a new flow it passes flow information to the QoS controller which assigns the correct radio link priority for the flow. Next WFMP establishes the radio flow with the allocated priority. WFMP allocates RAN_IDs and updates the access point RAN_ID – radio flow tables. The M-router includes a master WFMP which classifies the flows and maintains the data base of all the existing flows while the mobile terminal includes only a simple WFMP entity that multiplexes the RAN_IDs into the correct radio flows.

Mobile Control Protocol (MCP): MCP protocol transmits WFMP and MMC messages between the mobile terminals and the M-router. MCP provides a reliable mechanism for transmitting control information. MCP implements a simple go-back-N type of retransmission protocol. A separate low layer protocol was added to guarantee reliable transmission of control messages instead of using TCP/IP. This is because TCP/IP will not allow the separation of control messages from other TCP traffic. Therefore, e.g. in the case of handover the control messages would be mixed with the user data traffic, which causes a significant delay for the handover procedure and re-establishing connections. The use of MCP allows control traffic to be prioritised before the user data packets.

Flow Compression (FC) block: The M-router and mobile terminals include FC entities which compress the detected IP flows. Flow compression is used only for the classified IP flows. The other IP traffic is sent without compression.

Radio Resource Manager (RRM): Each access point has a RRM entity that manages the radio resources of the particular access point. In the WAND system WFMP sends resource queries to RRM each time a new flow is established. WFMP transmits the requested radio flow priority (allocated by QoS) and estimated flow rate (WFMP evaluator). Based on this information RRM decides whether the connection is accepted or not.

Access Point Control Protocol (APCP)**:** APCP protocol provides a mechanism for transmitting control messages between the access points and the M-router. APCP can be located above the TCP/IP stack which guarantees the reliable transmission of control messages. WFMP deploys APCP for RRM queries and to send flow control information to the radio sub-system.

12. FLOW MANAGEMENT

12.1 Outline of IP Flow Management

IP implements a connectionless packet data system. The data is carried inside packets, the header of which indicates the correct destination address. This transmission scheme does not enable the system to separate various connections. The only possible way to detect connections is to monitor the IP traffic inside the M-router and to detect and classify IP packet streams called IP flows. An IP flow is established if two hosts (applications) send IP packets frequently between each other, or if RSVP control messages are received. The network can assign certain QoS characteristics for a flow, which is essential for multimedia service implementation in IP networks. For instance a particular flow can be prioritised in the router.

The objective is to define a mechanism which maintains IP flow QoS characteristics in the air interface and allows different IP packets (flows) to be prioritised in the radio link. The defined concept deploys radio flows which are created between the mobile terminal and the access points. An overview of the proposed flow management approach is as follows:

The M-router monitors the headers of the incoming IP packets and tries to classify the existing IP flows (i.e. regular packet streams) utilising the IPv6 flow label and/or destination and source IP addresses and ports. If the M-router detects an IP flow, it will start marking the packets that belong to this flow with a specific RAN_ID (Radio Access Network IDentifier). The router allocates a unique RAN_ID for each detected flow. *The RAN_ID is utilised to separate packets belonging to IP flows in the radio access network*. Consequently, the use of RAN_ID will create “virtual connections” through which IP flows are packed across the RAN (MT-MR). RAN_IDs correspond to ATM VPI/VCI identifiers. A proprietary Wireless Flow Management Protocol (WFMP) is defined for managing the RAN identifiers. Both the terminal and the M-router have WFMP entities which communicate peer-to-peer over the wireless link. WFMP actually provides the convergence layer functionality. The M-router WFMP detects flows, allocates RAN_IDs and informs the MT WFMP of the assigned ID value. To minimise the overhead the 24-bit RAN_ID is compressed into an 8-bit MVC identifier in the radio link. The MVC scheme supports 256 flows per terminal.

The radio link is the bottleneck in terms of QoS provision and throughput. The proposed system includes two mechanisms to improve the wireless support for broadband services: priority queues and flow compression. The radio sub-system handles various radio flows differently. It has *three separate buffering queues for the incoming traffic*: *High priority queue for real-time traffic, medium priority queue for non-real-time data and low priority queue for best-effort data*. *Normally the M-router establishes flows, but the system also allows the mobile terminal to request the M-router to establish a radio flow (RAN ID) with a given priority for a particular IP flow.* As the M-router classifies the IP flow it assigns one of the three priority classes to the established radio flow. The decision is made on the basis of the IP flow type and traffic characteristics. The presence of multiple radio flow priorities enables the provision of wireless broadband services.

IPv6 headers are large (40 bytes for the base IPv6 header alone). The radio overhead is minimised by compressing the IP headers of detected flows. The compression is performed between the M-router and the mobile terminal. For this purpose these entities include specific Flow Compression (FC) entities. The IP header compression is efficient for flows as the IP source and destination can also be identified from the RAN_ID. The receiving end can look up the RAN_ID and decompress the IP header accordingly.

Only detected IP level flows will be switched into separate radio flows. The other IP packets are transmitted over the air in the default channels. The system defines three fixed (hard-coded) radio flow identifiers that will be used for transmitting non-flow packets. In this case IP packets from multiple different sources are multiplexed into the same radio flow (RAN_ID), making IP header compression impossible.

The system offers three multiplexed radio flows per terminal, one for each radio priority queue. These "default" radio flows were defined to improve the support for inter-IP domain mobility and differentiated services. In the defined scheme the M-router or the terminal can look at the priority bits of a single IP packet and send that with the corresponding radio QoS (priority). As soon as the IP flow is detected the packets will be switched into a separate radio flow with a specific QoS and with IP header compression. Figure 28 summarises the defined flow management concept.

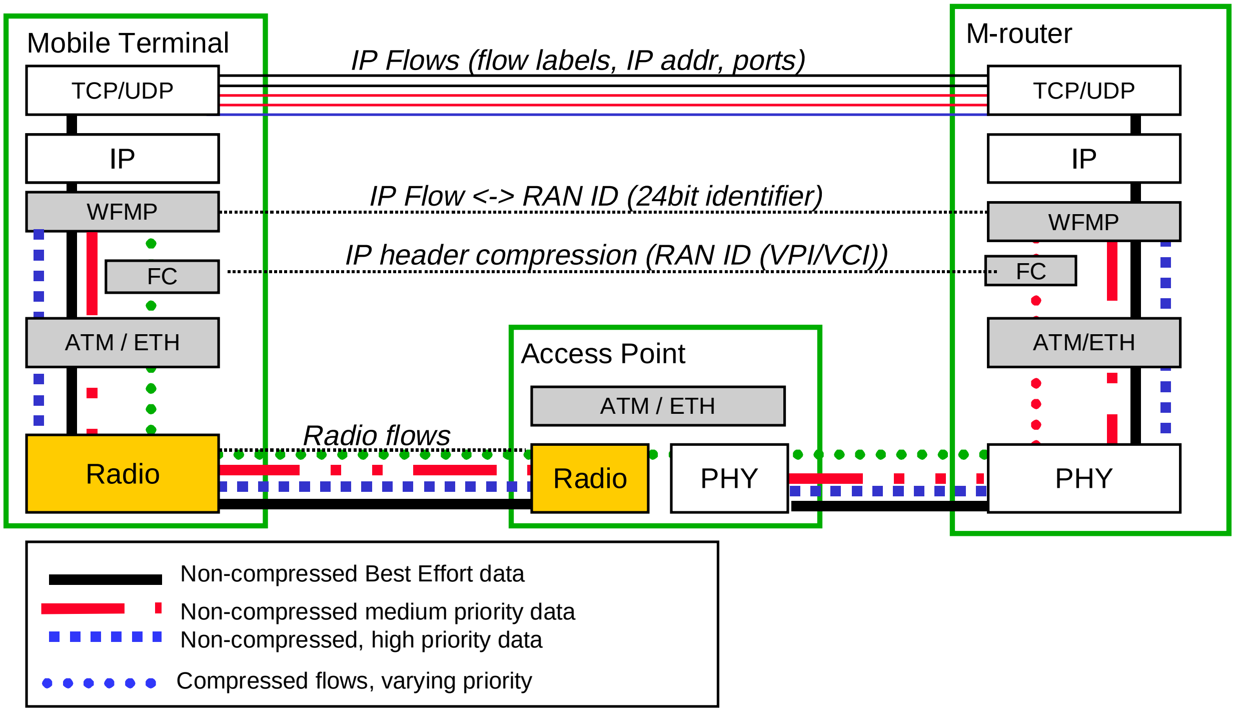

Mobile Terminal
TCP/UDP
IP
WFMP
FC
ATM / ETH
Radio
IP Flows (flow labels, IP addr, ports)
IP Flow <-> RAN ID (24bit identifier)
IP header compression (RAN ID (VPI/VCI))
Radio flows
Access Point
ATM / ETH
Radio
PHY
M-router
TCP/UDP
IP
WFMP
FC
ATM/ETH
PHY
Non-compressed Best Effort data
Non-compressed medium priority data
Non-compressed, high priority data
Compressed flows, varying priority

Figure 28: The Defined Flow Management Scheme

The detected IP flows are marked with a dedicated RAN_ID label (in ATM VPI/VCI) while the other IP data packets are sent using a default best-effort data RAN_ID label. The access point performs RAN flow – radio flow multiplexing, i.e. mapping between RAN_ID and MVC. The

M-router manages the location of MTs and updates the routing table if the mobile performs handover between access points.

12.2 Flow Classification

To provide different levels of QoS to different packets, there must be mechanisms to distinguish between these different sets of packets (traffic flows). Not only must different flows be identified, but it also must be decided when a given traffic flow should receive something other than best effort service. Another major issue is the level of granularity that should be employed, should QoS be provided on a per application flow basis (e.g. on the basis of a TCP or UDP port number), or should these flows be aggregated in some manner. This design note discusses these issues and presents alternative solutions.

12.2.1 IPv6 Flow Identification

In IPv6 all packets belonging to the same traffic flow must be sent with the same source address, destination address, flow label, and hop-by-hop options extension header. Furthermore, if the packet contains a routing header, each packet in that flow must have the same contents in all extension headers up to and including the routing header (note the hop-by-hop and destination extension headers are usually placed before the routing header) [Dee95]. If the flow label is non zero, then the combination of source address and flow label identifies a traffic flow. These three possibilities are summarised in Table 4.

Table 4: Specified IPv6 Flow Identifiers

Flow Type	IPv6 Header Fields
I	Source Addr; Flow Label
II	Source Addr; Dest Addr; Flow Label; Hop-by-Hop Options
III	Source Addr; Dest Addr; Flow Label; Hop-by-Hop Options; Destination Options; Routing Header

If the flow label is zero it provides no additional information about the flow. Similarly the hop-by-hop, destination and routing headers do not contain information related to details about the flow. Hence, from the three alternatives in Table 1, the two possible techniques for detecting a flow, on the basis of IPv6 header fields alone, appear to be:

- source address + flow label
- source address + destination address

It is also possible to employ additional criteria for identifying flows, such as transport protocol port numbers (thus allowing different application traffic to be identified). In addition, when we have port numbers in a flow id we also need the protocol id of the transport protocol. Thus a third flow detection options is:

- source address + destination address + transport protocol port numbers + protocol id

It is important to note that due to the extension header format of IPv6, significant processing time may be required to access some header fields (e.g. the TCP or UDP port number) since all preceding headers will need to be parsed first. Hence the feasibility of this third solution needs further consideration. We believe that for this reason the flow label option should be used when ever possible. A solution based on port numbers should only be employed when the flow label is zero.

12.2.2 Granularity of QoS Provision

Flows can be detected at a number of levels of granularity. The previous section described how flows can be identified on the basis of IPv6 source and destination addresses, non zero flow labels and transport protocol port numbers.

It is currently not clear what level of granularity will be provided via the flow label. For instance a different flow label could be selected to represent each possible transport protocol, or application operating between that source and destination. We assume that in the future hosts will select the flow label on an application basis.

It is also possible to select even coarser levels of granularity. For example all datagrams sent from, and or destined to the same subnet, or even all traffic with the same next hop on their routing path. This may be useful for virtual private networking services. The problem with this approach is that different applications running between two subnets may have very different QoS requirements. This approach does not allow them to be differentiated. Another way to group traffic is by service classes. For instance all traffic carrying the same transport protocol could be grouped into the same traffic flow, even if their source and destination addresses differ, at places in the network where they share the same path.

[Wor97] lists the following possible levels of granularity:

- source address and flow label - a single application
- source and destination address and ports - for example a single TCP connection
- source and destination address and destination port - designed for WWW traffic so that all traffic from a WWW server to a given destination is carried over the same flow (even if the source port differs).
- network routes - all traffic that shares a common route across a network
- route to egress routers - all traffic destined for the same egress router
- source and destination subnetwork addresses - all traffic travelling between two subnets.
- next hop routes - all traffic that shares the same next hop on their forwarding path.

In the wireless IP system the motivation for detecting flows is to assign them a suitable QoS at the radio level. Therefore the first three alternatives listed above are most appropriate. The final four options do not allow us to separate traffic on the basis of QoS requirements sufficiently. That is, the level of granularity is too coarse. For example if all traffic from a given company is grouped together, then if one user is running a real-time video application and another is WWW browsing their traffic would be treated in the same manner.

However the other danger is that if the granularity is extremely fine (e.g. each application flow is differentiated) scalability problems can arise because of the volume of state information that must be maintained.

An important aspect of selecting the appropriate level of granularity, is flow characterisation. That is examining the characteristics of the traffic flowing across the network, and from this select the granularity (and detection scheme) that best suits the observed traffic mix. [Cla95] examines the characteristics of individual traffic flows and how they can be aggregated together. Their three key findings were that (a) a significant percentage of traffic flows are short-lived even when different granularity levels are employed; (b) the number of host pair flows is not significantly greater than the number of destination network flows and (c) traffic caching schemes could benefit from using application information.

12.2.3 Flow Signalling and Flow Detection

In the Internet by default all IP traffic is carried with best-effort service. To enable the Internet to provide different levels of QoS, mechanisms are needed to decide when a given set of packets should be provided with non-best effort service. For instance in an ATM based system, all best effort IP packets would be carried on one VC. When a traffic flow is detected that needs better QoS a dedicated VC would be created for that traffic. This section describes different mechanisms that could be employed to detect flows.

If RSVP is employed by applications to request resources, the arrival of RSVP PATH and RESV messages could be used by the flow detector to indicate that a group of packets should receive a new level of QoS. RSVP RESV messages contain a *filter spec* that identifies the set of packets that should receive the desired QoS. In IPv6 the filter spec is given by either the IPv6 source address and source port, or, the IPv6 source address and flow label. RSVP also defines a *session* to be a data flow with a particular destination and transport layer protocol. Each RSVP session is defined by the triple: Destination Address, Protocol ID and optionally Destination Port, plus some RSVP specific. The session information is present in both PATH and RESV messages. Therefore, flows can be detected on the arrival of RSVP RESV messages. In this case all IP datagrams with header fields that match both the session and filter spec details, should receive the new level of QoS. In practise the RSVP messages can be decoded in the M-router to obtain explicit QoS parameters for the RSVP flow. This approach can be used in conjunction with two levels of granularity: one on the transport protocol basis (no destination port provided), and the other on the per application basis (destination port is provided).

An alternative approach to requesting QoS in the Internet is via Differentiated Services (diffserv). In this case, no special signalling messages are transmitted to indicate that a given set of packet should receive a certain QoS. Instead diffserv sets bits in the IP header to indicate the delay and dropping preferences of that datagram. The one octet traffic class field (termed the DS field) in IPv6 contains this information [Dee97]. If this field is zero the traffic should be carried in a best effort fashion. However, if the detector observes packets with non zero entries this could be used as a trigger to create a flow with a higher level of QoS.

RSVP tends to be used in a dynamic manner, that is, when an application requiring QoS begins it informs the routers across the network that resources should be reserved for that specific RVSP session. In contrast, DiffServ tends to be used to pre-provision QoS for each of the traffic class values the network operator wishes to support. This means that when the detector sees a datagram with a given traffic class value it does not need to trigger the creation of a VC or similar, but simply to note via which existing pipe or VC the packet should be transmitted. Hence in this case detection is on the basis of the contents of the traffic class field.

The three flow identifiers mentioned above do not necessarily need to be employed, because the detection decision is on the basis of the traffic class only. Thus this approach can be employed on the granularity of all packets requiring the same level of (regardless of source and destination). Indeed here dedicated flows are not required. Instead the diffserv QoS class can be mapped to the appropriate radio flow QoS and the packet sent using this QoS.

The previous paragraphs have described cases where QoS information is used to detect flows that need to be carried with non best effort QoS. However flow detection can also be based on traffic statistics. One example of this approach would be for the flow detector to measure the volume of packets with the same flow identifier. When this exceeds a pre-defined threshold, a special flow could be created for that traffic. The threshold could be the number of packets seen within a given period of time (e.g. if more than 5 datagrams with the same flow identifier are seen in 30 seconds). This is called the X/Y classifier and is an effective, but simple mechanism for detecting flows.

An alternative approach would be to assume that only packets with non zero flow labels should be selected as flows requiring a special QoS. That is, the detector only selects flows with a type I flow identifier. Again the flow could be selected if a certain volume of datagrams with this flow identifier are observed, or the detector could select the flow when it sees the first datagram with this flow identifier. This approach is based on the assumption flow labels are only employed for traffic with non-best effort traffic requirements. The problem with this approach is that it is very possible that many real-time applications do not employ the flow label, yet they want more than best-effort service. Another issue in this case is to decide what QoS to provide this flow, since no QoS information is provided.

If packets are grouped together on the basis of a flow identifier, then all packets between a given source and destination will be grouped together, and carried in the same manner. The operator may decide a finer level of granularity is more appropriate. For instance the detector could also monitor the next header fields to determine which transport protocol is employed (e.g. TCP, UDP), and only forward e.g. datagrams containing TCP packets in a non best effort manner. Alternatively the detector could employ an even finer level of granularity and monitor the transport layer port numbers, and only forward for example IP datagrams containing WWW or FTP traffic (i.e. those with a given port number) in a non best effort fashion.

An alternative approach is to treat traffic flowing in different directions between a given pair of hosts (e.g. a MT and a WWW server) separately. [Hou97] suggests that server (e.g. WWW server) originated traffic tends to involve large packets and hence should be detected as a flow as early as possible. This paper recommends that each server originated flow be detected and treated as a separate flow. However traffic from multiple hosts, destined for the same server should be aggregated together and treated as one flow, due to the much lower traffic volumes involved. Aggregating this low volume traffic is hoped to provide a better use of resources (at least by reducing the volume of state information maintained) [Hou97].

The choice of an appropriate flow detection algorithm is an implementation specific issue. The optimal flow detection scheme is highly dependent on the characteristics of the traffic being carried on a given network. A problem with creating different dedicated state (or ATM VCs) for individual flows (e.g. on the basis of RSVP messages, or IPv6 flow identifiers) is scalability.

Both Ipsilon (now Nokia Networks) IP Switching [New96],[New96b] and Toshiba's Cell Switch Router [Kat97], [Kat97b] approaches, support several of the flow detection approaches described above. These are the RSVP, traffic volume, and identification of specific higher layer protocols (e.g. WWW, FTP). As discussed in [Kat97] the motive for flow detection based on RSVP is to meet the QoS requirements of traffic, in particular over the radio link. The motivation behind the other two approaches is twofold: (a) to reduce the delay for high volume traffic, especially over the air interface, and (b) to reduce the IP level processing burden on the router.

The choice of flow detection technique is also related to the level of granularity required. As discussed above, many flow detection techniques can be employed at multiple granularity levels. Many approaches use data-driven flow techniques for example: IP Switching, CSR, IP/ATM [Par95] and IPSOFACTO [Ach97]. In this case flows are detected when data (or RSVP messages) arrive for that flow. The granularity of the flow can vary enormously.

In contrast Tag Switching [Rek97],[Rek97b] employs topology driven flow detection. In this case flows are identified based on routing information before any data arrives. This means a flow is able to be associated with a group of routes, a multicast tree, a source-destination pair, an application operating between a given source and destination, or any other policy [Rek97], [Rek97b].

Multiprotocol Label Switching (MPLS), currently being specified by the IETF, is required to support both data driven and topology driven flow detection techniques [Cal97]. Hence the choice of flow detection technique will become an implementation issue.

Another issue, closely related to flow detection, that must be considered is how to decide when to tear down or remove flow specific information. This could be based on a timer. In this case flow state information is removed when no packets matching the flow are seen over a specified time period. The difficulty in this case is selecting the value of the timer. This issue is considered in [Sar94]. RSVP flows can be torn down if RSVP update messages are not received, or if an explicit RSVP RESVTEAR message is received. Another approach would be to wait for the arrival of a message to indicate the flow is finished. For example the arrival of TCP FIN messages. The problem with this last approach is that it requires processing of higher layer protocols. Hence we prefer a timer based solution for all flows. However for RSVP flows the arrival of RSVP control messages can also be used to decide whether to terminate the flow.

12.2.4 Flow Classification Summary

The general problem of flow classification is a complex one. There are many ways that flows that require special service (i.e. non best effort service) could be detected. A good knowledge of the characteristics of the traffic being carried over the network, is essential to select an appropriate detection mechanism. In a wireless IP local area network, such as that considered by the WP1 extension, the volume of application flows will be considerably lower than in the core Internet. Hence scalability may not be as great an issue. The other issues that must be kept in mind in the WAND system are that:

1. In the scope of WAND / wireless IP broadband network the flow detection will be performed in the M-router. No assumptions are made of the fixed / core network.
2. Flow detection and selecting QoS are independent processes. Firstly the system has to identify which IP packets form a flow (enough traffic per time) and/or require specific QoS in the radio link. Secondly, the system has to be able to establish radio level flows and switch the detected IP flows into radio flows. Thirdly, the system has to include an independent mechanism that determines the proper radio QoS class for the established flows. The current assumption is that the radio sub-system could provide three different priority queues to which different IP flows will be mapped.
3. The developed scheme can utilise upper layer protocols (TCP/UDP, RSVP) but it should be based on a pure IP based mechanism that is only improved with upper layer processing.

In WAND, the motivation behind detecting flows, is to determine which flows should be provided better than best effort service. This will include traffic with real-time QoS requirements, but also long-lived traffic flows such as WWW, or FTP traffic. As stated in [Sta98], the WAND wireless IP system should operate with both RSVP and DiffServ. Hence the flow detection approach must support both of these protocols. Therefore detection based on the arrival of RSVP messages, or IP datagrams with diffserv bits set is desirable. Due to the desire to provide better than best effort QoS to long-lived flows, it may also be beneficial to add a packet volume based detection scheme to the WAND flow detector. This may also help to obtain an estimate of the required radio resources. Essentially, the WAND flow detector should be sufficiently flexible it can detect flows on the basis of a variety of criteria.

The other issue that must be considered is the level of granularity, or aggregation that should be employed in the WAND system. This is hard to determine without knowledge of the volume and characteristics of the traffic being carried in such a network. If the volume of simultaneous application level traffic flows across the network is small, it may be feasible to create a different association for every application flow. The capabilities of the underlying scheduler must also be considered. For instance in the extreme if the scheduler was a simple FIFO mechanism there would be no value in separating the traffic into different flows. Thus the problem of selecting an appropriate scheduler and the appropriate level of flow aggregation are closely related. A solution that takes the capabilities of the scheduler into account is described in Section 12.4.

As discussed in the previous section, once an IP level flow has been detected it will be marked with a dedicated RAN_ID label. All other IP traffic will be send using the default best-effort RAN-ID label.

12.3 Packet Compression

In a wireless environment, the bandwidth of the air interface is a scarce network resource. Hence, it is important to minimise the volume of traffic being transmitted over the air interface. This can be achieved via compression. Two forms of compression are possible: (a) IP (and transport) header compression where a session is identified and all header fields that remain constant can be removed, and (b) payload compression where a lossless IP packet compression algorithm is applied to the contents of a packet.

Compression is even more important when carrying IPv6 datagrams, compared to IPv4 datagrams due to the significant increase in header size (the IPv6 base header alone is 40 octets). However it is also important to keep in mind that the bandwidth savings of a given scheme are a trade-off against the processing requirements on the mobile terminal (MT) (i.e. the complexity of the compression algorithms must be considered).

12.3.1 IP Switching Approach

IP Switching proposes several encapsulation schemes that reduce the size of the IP header transmitted. This makes the transmission of IP traffic more efficient. However the motivation behind the encapsulation schemes is security, to prevent traffic with different header information using the resources reserved for the real flow [New96b]. This is achieved by not transmitting unnecessary IP header fields [New96].

The current IP encapsulation types are defined for IPv4 networks only. These encapsulation types are linked to the flow identifiers. All IP packets in the same flow must have the same values for several header fields. Thus there is no need to transmit these header fields in every packet. Instead, they can be reinserted into the packet at the end-point of the dedicated VC. For the three IP Switching IPv4 encapsulation types, the savings per packet are 8, 24 and 16 bytes depending on the IP flow type.

The same concept can be applied for IPv6 based systems. Three flow identifiers are proposed, type I for use with non-zero flow labels (IP flows) and II and III for flows with zero flow labels (see Table 4). For type I and II flows all packets belonging to the flow must have the same values for the following fields: source address, destination address, flow label (in type II this is zero), and hop-by-hop options. Thus, these fields can be removed over the wireless link, creating significant resource savings. As stated above, for type III flows, all extension headers up to the routing header must have the same value. All fields in these headers can be removed from each packet. Thus for type III flows even greater bandwidth savings can be made.

To summarise, in IPv6 networks, encapsulation schemes are even more important because of the increased size of IPv6 headers compared to IPv4. However, it is important to note that the proposed encapsulation can only be employed for dedicated flows (i.e. flows that have an independent ATM VC created for them). For the best effort data we have to transmit intact IP headers since multiple traffic flows are carried on the same radio flow. Thus the header information is required to distinguish between these flows.

This technique scales well into our proposed architecture – header compression is only performed for detected flows and compressed IP packets are detected on the basis of RAN flow ID (e.g. VPI/VCI value).

12.3.2 IPv6 Header Compression

Within the IPng IETF working group there has been a work item to consider how to 'abbreviate' the IPv6 header. An internet-draft, [Deg97], describes compression methods that can be applied to IPv6 base and extension headers, IPv4 headers, TCP and UDP headers, and encapsulated IPv6 and IPv4 headers. Using these methods headers of typical UDP or TCP packets (normally 8 and 20 octets) can be compressed down to 4-7 octets including the 2 octet UDP or TCP checksum.

Note without header compression, the smallest possible IPv6 header size is 40 octets. Hence minimum IPv6/UDP and IPv6/TCP headers are 48 and 60 octets respectively. This does not include any IPv6 extension headers. In a mobile environment, it is likely that IPv6 packets will require at least a routing header or home address option header due to the need for Mobile IP. The length of these headers are 24 and 20 octets respectively.

Header compression relies on the fact that many header fields remain the same over the life-time of a traffic stream (the IP switching approach also relies on this). Fields that do not change between packets do not need to be transmitted. Other fields that change often, but in a predictable manner (e.g. sequence numbers) can be encoded incrementally, reducing the number of bits that must be transmitted. Only fields that change constantly in a random fashion need to be transmitted in every packet.

To initiate header compression, a full header (containing a compression identifier) is transmitted over a link. The compressor and de-compressor store those fields that are constant or change in a predictable manner as compression state. From then on packets can be transmitted over a link in a compressed format. Any change in fields that are expected to remain constant will cause the compressor to transmit a new full header, allowing the de-compressor to update its compression state.

As long as the compression state is the same at the compressor and de-compressor, headers will be de-compressed correctly. However if a packet containing a full header or compressed header is lost or damaged the compression state at the de-compressor is incorrect. Thus header compression methods must provide mechanisms to update the state in the de-compressor, and to detect or avoid incorrect de-compression.

Since TCP headers are compressed using the difference from the previous TCP header (due to the presence of sequence numbers), loss of a packet with a compressed or full header will cause subsequent compressed headers to be decompressed incorrectly. [Jac90] describes mechanisms for recovering from such loss. The situation is more difficult for non-TCP packet streams (i.e. it is harder to detect if a packet has been de-compressed incorrectly).

Incorrectly decompressed headers of non-TCP packets are not as well protected by checksums as TCP packets because differential coding is not used and there are no sequence numbers. To safely avoid incorrect decompression of non-TCP headers, each version of the compression state for each streams is identified by a generation, a small number that is carried by the full and compressed headers. The generation changes only when the context of a full header is different from the context of the previous full header. This means that losing a full header or compressed headers will make the context of the de-compressor obsolete only when the full header would actually have changed the context. In the WAND system FEC is employed in the air interface and also ARQ for non real-time traffic. This will keep IP packet loss rates low in the RAN, hence situations where the de-compressor context becomes obsolete should be rare.

When a decompressor sees that a compressed header carries a generation value other than the generation of its compression state for that packet stream, the compression state is not up to date and the packet must be discarded or stored until a full header establishes correct compression state.

The generation field is 6 bits long so the generation value repeats itself after 64 changes to the compression state. To avoid incorrect decompression after error bursts or other temporary disruptions, the compressor must not reuse the same generation value after a shorter time than MIN_WRAP seconds. Moreover, a de-compressor which has been disconnected MIN_WRAP seconds (set to 3 seconds in [Deg97]) or more must wait for the next full header before decompressing.

To allow the decompressor to recover quickly from loss of a full header that would have changed the context exponential back-off of full header refreshes can be employed. In this case full headers are sent periodically with an exponentially increasing period after a change in the compression state. Furthermore, to avoid losing too many packets if a receiver has lost its context, there is an upper limit, F_MAX_PERIOD (default value of 256), on the number of non-TCP packets with compressed headers that may be sent between header refreshes.

To further improve compression efficiency, a number of packets can be grouped together into a packet stream for compression. The packets should be grouped in such a way that packets in the same packet stream have similar headers. If this grouping fails, thrashing may occur as the compression algorithm can rarely utilise the existing compression state for the packet stream and full headers must be sent frequently. In the WAND system flows are detected by the M-router and carried as dedicated radio flows on the basis of constant header field values. Thus header information for this flow will not change (if it did it would be carried on a different radio flow). Hence thrashing of the compression algorithm should not occur, for dedicated flows at least. The value of compression for default traffic is not clear, since in this case IP packet headers will change regularly.

Grouping is done by the compressor. A compressor may use whatever criteria it desires to group packets into packet streams. For more details see [Deg97]. A compressor is also free not to group packets into a packet stream for compression, letting some packets keep their regular headers and passing them through unmodified.

As long as the rules for when to send full headers for a packet stream are followed and the sub-headers are compressed as described above, the decompressor is able to reconstruct a compressed header correctly regardless of how packets are grouped into packet streams.

It is important to note that these compression methods are only applicable for point-to-point links. The mechanisms described here are intended for a point-to-point link. Care has been taken when extending these methods for multi-access links and multicast.

As discussed above additional information needs to be added to full headers or compressed headers. For TCP packet streams this is 2 octets. For non-TCP packets this can be up to 4 octets. The detailed formats are given in [Jac90] and [Deg97].
For IPv6 the entire base header can be compressed away, because all fields either do not change or can be inferred for a given packet stream.

For a given traffic flow, which IPv6 extension headers are present and the relative order of them is not expected to change. Whenever there is a change, a full packet header must be sent. This also means that all next header fields do not need to be transmitted. Details for how each IPv6 extension header can be compressed are given in [Deg97]. In the case of handover sending of a full header is also mandatory due to the change in where the flow is forwarded, particularly for inter-M-router handover.

UDP headers can typically be reduced to 2 octets containing the UDP checksum. For TCP the compressed size varies depending on whether differential coding is employed. The following hook is supplied to allow additional header compression schemes for headers on top of UDP. The initial chain of sub-headers is compressed as described in [Deg97]. An additional header compression scheme such as Compressed RTP [Cas97] could be employed for higher layer headers.

From this description it seems that even greater savings can be made by employing IP header compression as described in [Deg97], rather than IP switching encapsulation.

12.3.3 IP Payload Compression

The IETF IP Payload Compression Protocol Working Group is developing protocols that allow lossless compression to be performed on individual payloads before the payload is processed by a protocol that encrypts it (if any). The primary goal of the working group is to develop compression schemes that can be employed in conjunction with IPSec over IPv4 or IPv6 networks. The availability of compression protocols that will inter-work with security protocols will become even more important with the increased commercialisation of the Internet, and security consciousness of its users.

Compression at the IP payload level is especially valuable when encryption is applied to IP datagrams. This is because encrypting the IP datagram causes the data to be random in nature, making if difficult to significantly compress PDUs at lower protocol layers. Thus, if both compression and encryption are required, compression must be applied before encryption. Payload compression can also be dangerous if media compression has already been applied. This can often be the case for video and audio streams. Compressing an already compressed payload can actually enlarge the payload. Thus care must be taken when deciding whether to compress a given payload.

[Sha98] provides the framework of the IP payload compression protocol (IPComp). Specific compression algorithms and formats using either LZS [Fri98] or DEFLATE [Per98] have been published as companion Internet Drafts. IPComp provides stateless compression, i.e. each payload is compressed independently.

In IPv6, the compression of outbound IP datagrams must occur before the addition of either a Hop-by-Hop Options header or a Routing Header. This is because both of these headers carry information that must be examined and processed by possibly every node along a packet's delivery path, and therefore need to be sent in their original form. Thus, IPComp is viewed as an end-to-end payload, and must not apply to hop-by-hop, routing, and fragmentation extension headers. That is, the compression is applied starting at the first IP Header Option field that does not carry information that must be examined and processed by nodes along a packet's delivery path and continues to the upper layer protocol payload of the IP datagram (i.e. the compression covers the payload of the IP datagram). It is important to note that the base IPv6 header (and the extension headers mentioned above), are not compressed. The only changes needed to the IPv6 header are to the payload length and next header fields. This type of compression scheme is not relevant in the wireless IP system for two reasons:

1. This compression should be performed end-to-end between communicating hosts – not between the MT and M-router.
2. If the data is encrypted the M-router cannot perform IP packet compression.

To summarise, IP payload compression can be performed between a MT and other hosts but it is transparent to the APs and M-routers..

In a wireless IP system such as that being specified in WAND, it may also be possible to employ some sort of header compression between the mobility enhanced router and the MT in addition to IPComp. This would provide even further bandwidth savings over the wireless interface, compared to header compression alone. However, the presence of IPcomp is negotiated on a host - host basis, and is thus out of the scope of the wireless IP system.

12.3.4 Summary of Compression Schemes

This section has described three approaches for reducing the size of IP datagrams. The third approach does not compress the IPv6 base header, instead it compresses any end-to-end extension headers and the IP datagram payload. The benefit of this approach in terms of bandwidth savings is hence very dependent on the contents of the datagram. Furthermore its use is out of the scope of the wireless IP system since it is an end-to-end protocol.

In the first approach, IP header fields that are constant for all packets in a data stream are removed. This can provide significant bandwidth savings, especially if the IPv6 datagram contains multiple extension headers. The drawback of this scheme is that it is specific to IP switching. That is, it is not possible to send datagrams modified in this way via routers that do not understand the IP switching protocol IFMP. The major motivation for compression in the WAND system is to minimise the size of packets transmitted over the wireless link. Hence it does not appear to be a major disadvantage the M-router communicates to the fixed network using complete IP headers.

One benefit of the second header compression approach is that this is more likely to gain wider use in the IPv6 backbone. If IPv6 header compression is employed in the backbone, this will reduce the compression processing requirements on the M-router, because it will receive packets already compressed. Another benefit of the IPv6 header compression approach is that fields that change value in a predictable manner (such as sequence numbers) can also be compressed. This is not possible in the IP switching approach. Thus we recommend the use of the IPv6 header compression approach.

12.4 IP Flow Multiplexing

Figure 29 and Figure 30 illustrate the flow management procedure for the downlink traffic. The process is as follows:

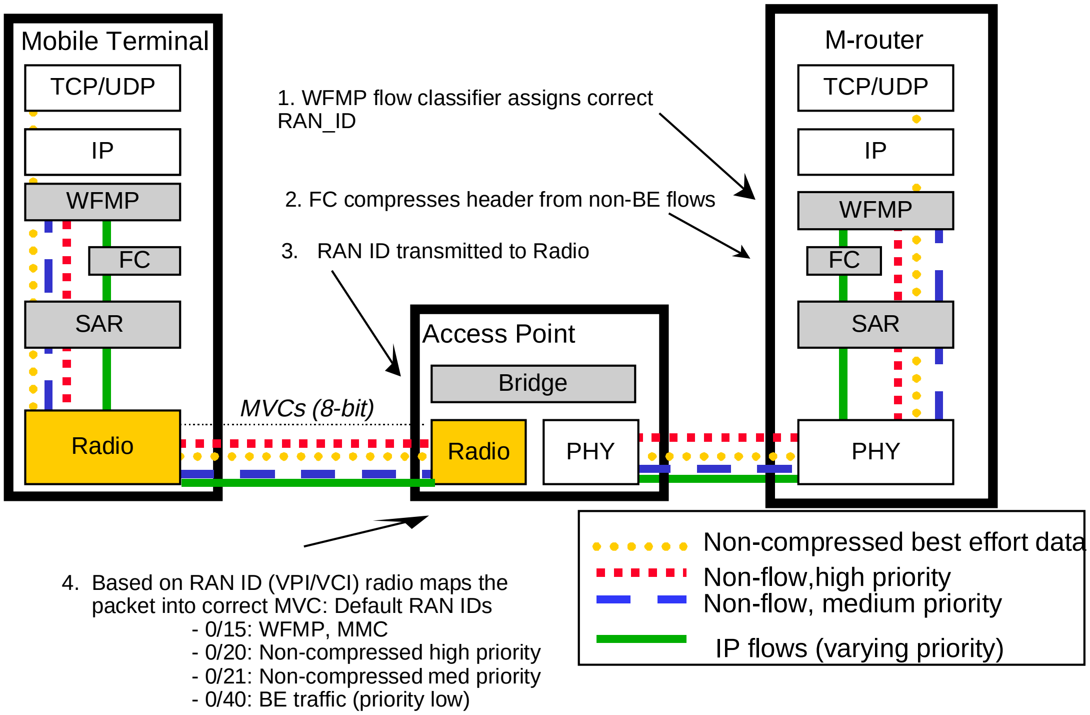
The MR-WFMP monitors the incoming traffic continuously. As the amount of packets per IP flow (between certain hosts (ports)) exceeds the threshold value (per time), WFMP establishes a RAN flow and allocates a new RAN_ID for it. Next the packets belonging to the flow are passed to the access point via Flow Compression (FC) using the allocated flow specific RAN_ID. The IP packets that do not belong to any flow are marked with one of three default RAN_IDs.

The FC entity compresses the IP header of the detected flows and copies RAN_ID to the resulting packet. At the receiving end the peer FC entity can detect the correct source by decoding the RAN_ID and decompressing the IP header before the packet is passed to the upper layers. Only detected IP flows are compressed.

The access point has a conversion table which maps the MVCs (radio flows) into correct RAN_IDs. The M-router allocates RAN_ID address space per access point. The packets are next transmitted to the M-router with RAN_ID “flows”. The packets are next transmitted to the radio link. The default RAN_IDs are assigned with a fixed radio link priority. Three default “pipes” (RAN_IDs) exist for user data: real-time flow, non-real time flow and best-effort flow. Each has a pre-configured RAN_ID. The non-flow IP traffic is transmitted within these flows without any compression.

The radio of the access point (layers 1 and 2) allocates unique MVC values per RAN_ID. Three default MVC “pipes” exists for the non-compressed traffic also in the air interface. The default RAN_IDs are mapped into the corresponding hard-coded MVCs while compressed IP flows are switched into dedicated MVC connections. Different flows are separated in the air interface using MVC and terminal wireless MAC addresses.

In the MT the radio modem converts the received MVC into the corresponding RAN_ID value and passes the packet to the SAR layer that reassembles the data into IP packets still maintaining the RAN_ID information. The compressed flows are then passed to FC which identifies the RAN_ID and decompresses the IP packet. The default RAN_ID traffic is passed directly to WFMP.



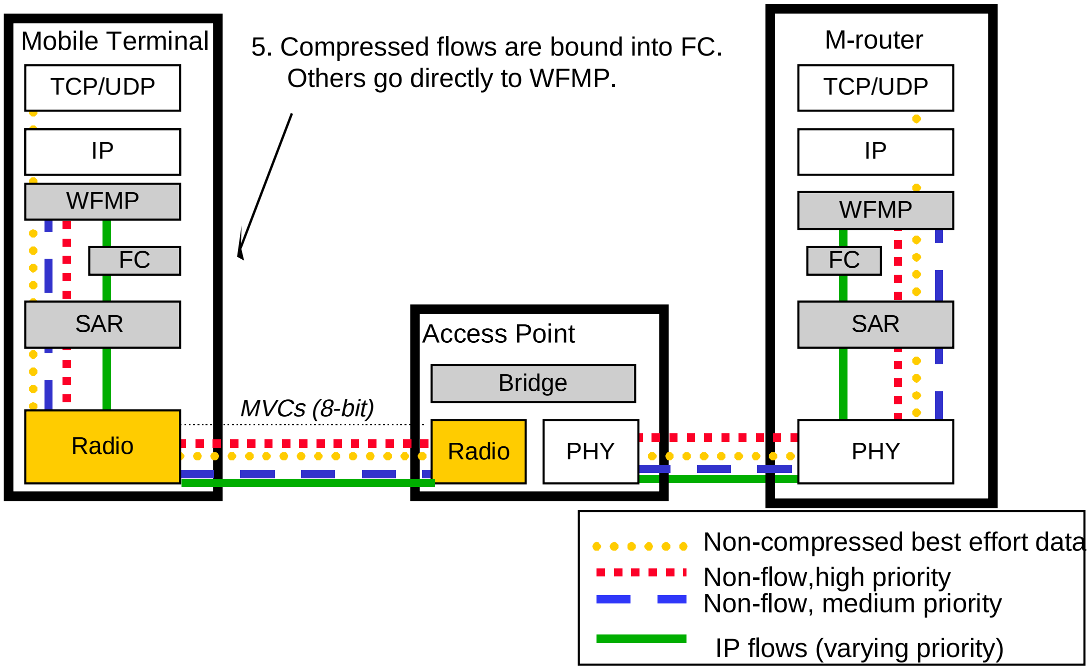
1. WFMP flow classifier assigns correct RAN_ID

2. FC compresses header from non-BE flows

3. RAN ID transmitted to Radio

4. Based on RAN ID (VPI/VCI) radio maps the packet into correct MVC: Default RAN IDs
   - 0/15: WFMP, MMC
   - 0/20: Non-compressed high priority
   - 0/21: Non-compressed med priority
   - 0/40: BE traffic (priority low)

Figure 29: IP and Radio Flow Multiplexing Scheme - Part I



5. Compressed flows are bound into FC. Others go directly to WFMP.

Figure 30: IP and Radio Flow Multiplexing Scheme - Part II

13. QOS MANAGEMENT

The WAND-system is designed to take advantage of customer and core network QoS mechanisms. In the large scale, Integrated Services based mechanisms are seen as QoS mechanisms in the last hop of the network. Differentiated Services based mechanisms are seen more as core network mechanisms.

In practice, QoS means differentiating classes of data service - offering network resources to higher-precedence service classes at the expense of lower precedence classes. QoS also means attempting to match the allocation of network resources to the characteristics of specific data flows.

13.1 Outline of the QoS Management Mechanism

In the wireless IP system QoS can be implemented by differentiating data flows on the basis of different information: IPv6 flow label + source address + destination address; port information + source address + destination address; priority bits + source address + destination address; or RSVP reservations. These flows can be treated differently from each other, and QoS can be implemented by multiplexing these flows on the basis of the QoS parameters of each flow. These parameters can be explicit values (peak cell rate, bandwidth requirement etc.) or simply information about the preferred Class of Service. Each case depends on the mechanism employed to determine the QoS parameters.

Packets belonging to a flow are placed in the appropriate radio queue. There are three different queues: Best Effort, Controlled Load and Guaranteed Service. These flows get different priorities from each other, and scheduling inside the queues will also be performed.

13.1.1 QoS Management Entity

The QoS manager’s main task is to map the fixed network’s QoS parameters to radio QoS and communicate with the radio resource manager. In practice this means mapping explicit QoS values to radio priority queues. The QoS manager has to know some statistics of the flow, and proportion this to the available radio bandwidth. With this information QoS manager can prioritise different flows.

QoS Manager has interfaces to RSVP and WFMP entities (see Figure 27). These interfaces and the main signalling messages are presented in Table 5.

MR’s QoS manager has more functionality than the MT’s QoS manager, because flow establishment is performed at the MR.

Table 5: QoS Manager Interfaces

Interface	Signalling Message
MR_QoS –>MR_RSVP	RESV_FLOW_conf
MR_RSVP –> MR_QoS	RESV_FLOW_req
MR_QOS –> MR_WFMP	RESERVE_FLOW_req
MR_WFMP –> MR_QoS	RESERVE_FLOW_conf
MR_QOS –> MR_WFMP	RR_STATUS_enquiry
MR_WFMP –> MR_QoS	RR_STATUS_reply
MR_QOS –> MR_WFMP	UPDATE_req
MR_WFMP –> MR_QoS	UPDATE_conf

13.2 Methods to Obtain QoS Information for a Flow

13.2.1 RSVP

RSVP is a resource reservation protocol, which tries to reserve bandwidth and desired QoS for a particular data flow. This system can be supported easily, because flows are detected by WFMP, and flows can be treated differently from others in the radio link. RSVP uses control messages trigger the reservation of resources in intermediate network elements. These control messages are separate from application data. These messages are separated from other data on the basis of the protocol number, and directed to the MR RSVP entity before the WFMP process.

We have two different scenarios when using RSVP:

1. WFMP has already detected a flow, and created a dedicated channel for the flow before the RSVP entity receives the reservation request for that particular flow.
2. RSVP entity receives the reservation request before WFMP detects the flow. In this latter case RSVP should trigger WFMP. This can be done via the QoS Manager.

Figure 31 shows how RSVP reservations are handled. This particular picture presents the situation where the MT is the receiver and the sender is somewhere in the network (downlink case).

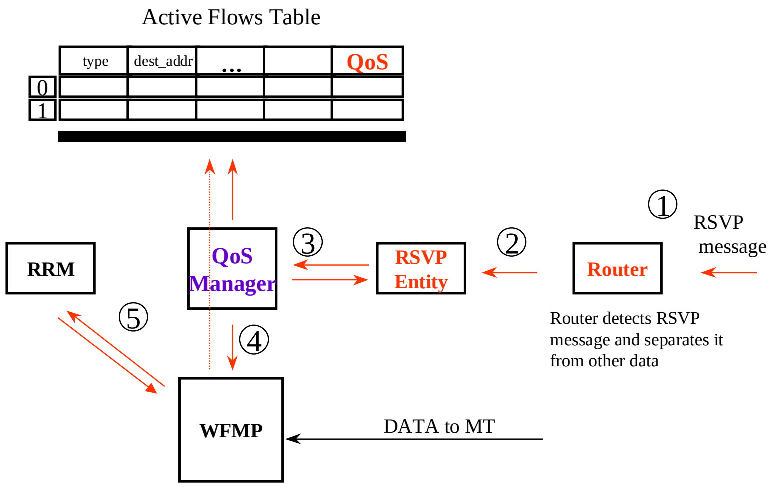


Figure 31: Obtaining QoS Information from Integrated Services

(1) RSVP messages (PATH/RESV) use protocol number 46, and that is how reservation messages can be separated from other traffic in the M-Router. (2) These messages will be delivered to the RSVP entity, which handles the messages on the basis of message type. (3) RSVP entity talks with the QoS manager which calculates the right priority class for the connection. (4) QoS Manager asks WFMP to establish a flow with the appropriate QoS. (5) WFMP asks for resources from RRM, and thus the QoS Manager doesn't need to do this.

In the M-router, the RSVP Entity has two roles: it acts like a normal RSVP Daemon, but also performs wireless specific operations. A normal RSVP Daemon checks the capacity of the M-router itself, and forwards / manipulates RSVP-messages at the IP level. In contrast the wireless RSVP Daemon communicates with WFMP and asks it to establish flows with specific QoS parameters.

RSVP PATH and RESV messages are sent periodically and thus can also be considered as refresh messages. These messages shouldn't trigger a new flow, but only refresh the existing flow information. This is done in MR-WFMP in the following way:

1. RESV refresh message triggers the QoS manager to send RESERVE_FLOW_req to WFMP.
2. WFMP checks from the active flows table if it already has signalled a flow for that particular data flow. In the Active Flows table, there has to be an indication if the flow is signalled by RSVP or detected on some other basis (like traffic volume).
3. If the flow already exists, WFMP sends a confirmation message, and performs no other actions.

13.2.2 Differentiated Services

Differentiated services means the deployment of priority bits in the IP-header to indicate the QoS requirements of the packet. As shown in Figure 32: (1) When data packet arrives with priority bits set (2) WFMP informs the QoS Manager about these bits if packets are detected regularly enough. (3) QoS Manager includes functionality that understands the bits, and maps these bits to the appropriate Radio QoS class. (4) WFMP Parameters are marked in the Connection table for that specific flow.

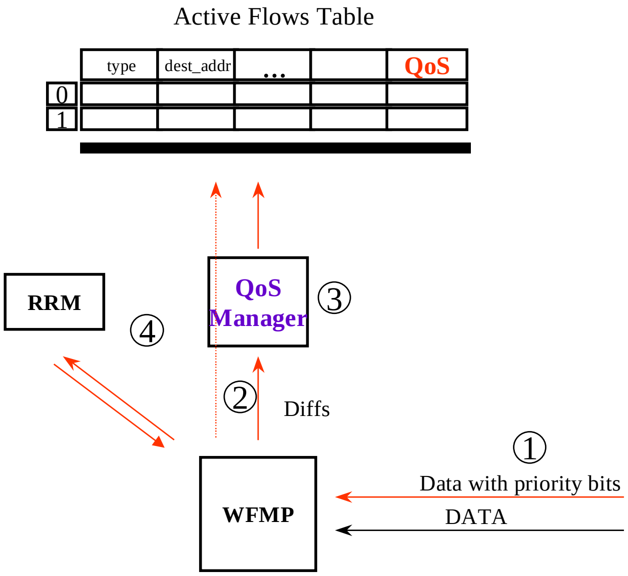


Figure 32: Differentiated Services

How parameters are mapped into the explicit QoS requirements, depends on the deployment of differentiated services on the backbone network side. When the meaning of the priority bits is well defined, they can be mapped into explicit Priority Classes. The standardisation of priority bits is still underway within the IETF, and there may be different ways to deploy priority bits in the future. The following tables show a rough example of how bits could be mapped into priority classes.

Table 6: Example of bit pattern

Bits	**Indication**
Bits 0-2	000 = Drop Preference 1, 001 = DP 2, … ,111 = DP 8
Bit 3	0 = Normal Delay, 1 = Low Delay
Bit 4	0 = Normal Throughput, 1 = High Throughput
Bit 5	0 = Normal Reliability, 1 = High Reliability
Bits 6-7	Reserved for Future Use

Table 7: Example of mapping Priority bits to QoS Classes

Priority bits	**QoS Class (**Table 9**)**
???001??	Class 3, BE
???101??	Class 2, Controlled Load
???110??	Class 1, Guaranteed

If WFMP detects packets that include priority bits, but can’t detect a flow from that traffic stream (not enough packets per second), WFMP should not give any special treatment for these packets. Otherwise, in the case of huge volumes of occasional packets, WFMP might get overloaded.

13.2.3 Well known ports

There are many “well known” TCP /UDP ports, indicating that traffic needs some real time features, or it may also indicate the likely volume of traffic. These kinds of ports are e.g. ftp-port or telnet ports, can have very different characteristics. FTP needs a lot of bandwidth, but doesn’t have critical real time requirements. On the other hand, telnet doesn’t need much bandwidth, but is adversely affected by high delay. This information can be employed when choosing the right radio link queue for the data flow. In the wireless IP system, it’s relevant to take advantage of port information after a flow has already been detected. This means that port information itself doesn’t trigger WFMP to notice a new flow, but after a flow has been detected, the port is taken into account. Clearly, if IPSEC or some other protocol hides port information, then it can’t be employed.

The next picture presents a situation, where WFMP has detected a flow and the port number belongs to an application that is identified by QoS Manager.

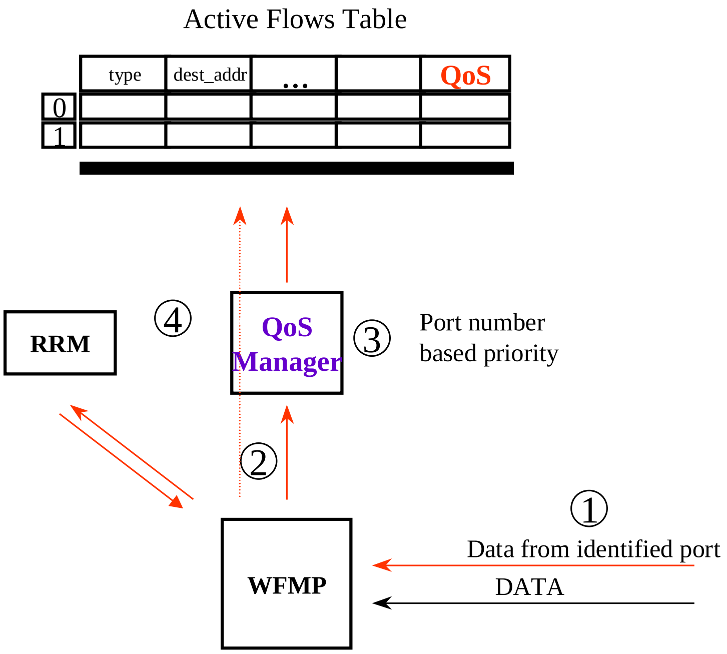


Figure 33: Port Information

Table 8 includes some common ports that could be treated differently. The listing of ports is only an example, and the ports that get specified service, will change over the time. It is important to note that it is desirable to treat ftp as a separate flow due to its long lived nature. However it is difficult to detect FTP data flows since FTP reassigns port numbers for data transfer.

Table 8: Ports (Example)

Type	**Port number**	**Explanation**	**Possible QoS Classes**
ftp-data	20/tcp	File Transfer [Default Data]	Best Effort
ftp-control	21/tcp	File Transfer [Control]	Best Effort
telnet	23/tcp	Telnet	Controlled Load
http	80/tcp	World Wide Web HTTP	Controlled Load
snmp	161/tcp	SNMP	Best Effort
ipx	213/tcp	IPX	Best Effort
dhcpv6-client	546/tcp	DHCPv6 Client	Controlled Load
dhcpv6	547/tcp	DHCPv6 Server	Controlled Load
vat	3456/tcp	VAT default data	Guaranteed
Vat-control	3457/tcp	VAT default control	Guaranteed

A question that needs to be considered is whether the network administrator should be able to configure ports that receive special handling. It may be desirable to configure the classification on the basis of what the customer company needs. Some companies may use multimedia applications much more aggressively than others. Also, some companies may use their own applications that should get most of the bandwidth (e.g. banks). These kind of special treatments are possible, if the QoS Manager is a separate functional entity that can be updated easily.

13.3 Radio Link QoS Functionality.

The QoS based radio access network has to provide bandwidth on demand, class based queuing and reliability. In wireless transmission links multiplexing of different services into the medium requires consideration on four QoS accounts: bandwidth, delay, jitter, and reliability.

Bandwidth is the first requirement for QoS driven services i.e. to support the requested traffic parameters. In the wireless link the main objectives are efficient channel utilisation while maintaining service specific QoS for IP traffic. This means that the AP Scheduler should know the requested average and/or peak bandwidth of those connections for which the radio flow is to be established. This way the Scheduler can guarantee the satisfaction of bandwidth on demand and perform statistical multiplexing.

Delay and *Jitter* are primarily affected by the traffic scheduling over the wireless link. In the wireless IP approach the flow based connections are queued separately (queue for each connection) and connections are grouped into 3 different delay class queues. In order to place the packets in the right queue, the Scheduler (or queuing function) needs to know the flow ID and delay class of the incoming packet. Also, to take the delay and jitter requirements into account in choosing the packets to be sent, the Scheduler should know (a) the maximum allowed delay of the packets at RAN layer, and (b) keep a time stamp for each packet.

Reliability over the wireless link requires error control which is typically provided via coding and/or data re-transmission. Coding is used both for error detection and correction which imposes constant overhead over the applied data. ARQ (Automatic Retransmission reQuest) is only applied for corrupt packets. This is feasible as long as the packet loss probability is not too high and the retransmission delay is admissible. The scheduler needs information about ARQ usage per radio flow (connection) basis. (No ARQ, Limited ARQ, ARQ). FEC usage can be fixed, used for all packets.

Table 9 presents the mapping from IP level QoS into the radio access network specific QoS. The first two columns specify radio access queuing and error control while columns 3-5 show different IP level QoS concepts.

Table 9. Example of network QoS mapping into radio access QoS.

Delay Class	*Radio Access QoS*	*Transmission Protocol*	*Integrated Services*	*Differentiated Services*
1st class	*No ARQ+FEC*	*UDP/RTP flow*	*Guaranteed*	*low delay/high dropping*
2nd class	*Limited ARQ+FEC*	*UDP/RTP flow*	*Controlled Load*	*medium delay/ medium dropping*
3rd class	*ARQ+FEC*	*TCP flow/No Flow*	*Best Effort*	*high delay/low dropping*

The M-router will function as the central intelligence point of the radio access network detecting flows, classifying them and mapping network QoS concepts into radio QoS capabilities.

13.3.1 Queuing

The queuing strategy for priority classes 1 and 2 is based on the radio flows such that each radio flow has its own queue. Based on the flow ID, the right priority class can be chosen as well as the queue where the packet is placed. This approach is needed because the Scheduler has to be able to differentiate the connections and their QoS requirements. For Best Effort data (priority class 3), the flows may also be identified.

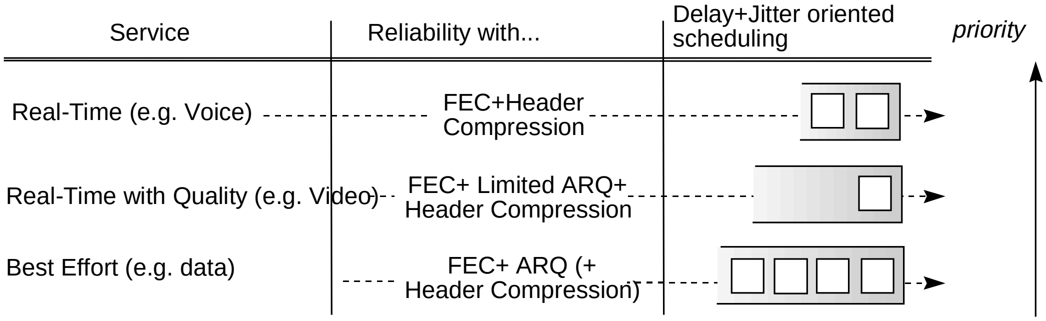

Service
Reliability with...
Delay+Jitter oriented scheduling
priority
Real-Time (e.g. Voice)
FEC+Header Compression
Real-Time with Quality (e.g. Video)
FEC+ Limited ARQ+ Header Compression
Best Effort (e.g. data)
FEC+ ARQ (+ Header Compression)

Figure 34: Wireless QoS Driven Queuing and Error Control Strategy

13.3.1.1 Simulation results

Delay and *Jitter* are primarily affected by the error protection scheme and traffic scheduling over the wireless link. We’ll study here what happens as Internet voice (UDP/IP traffic) and data (TCP/IP traffic) are scheduled into the air interface with service specific reliability but only with FIFO (First In First Out) queuing, which is the policy implemented by traditional Internet routers.

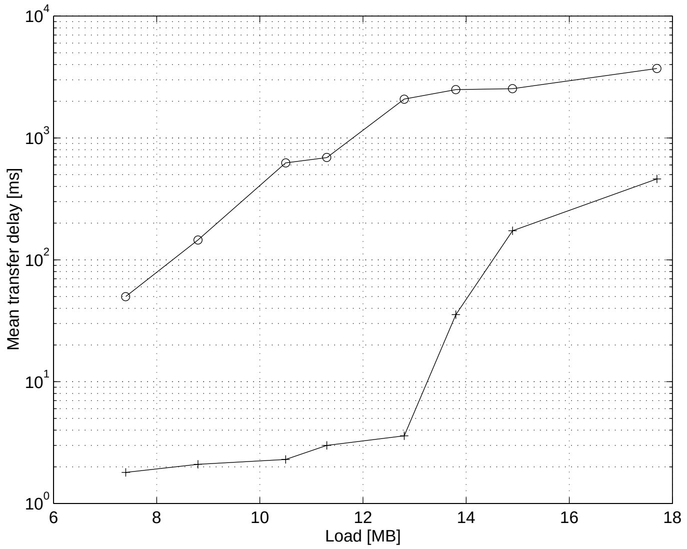


Figure 35: One service queue for Internet Voice (lower line) and Data (upper line)

Figure 35 shows that 10 Mb/s is the breaking point for ARQ based TCP/IP operation. Once the load is above 10 Mbits/s TCP/IP delay exceeds 500 ms. Voice over UDP exceeds its maximum delay (20 ms) at 13.5 Mbits/s. These points give us an indication of the load we can support. The second main observation is the fact that service specific delays seem to be correlated.

The required number of queues is typically tightly coupled with the service differentiation or link sharing policy adopted in fixed networks. We'll apply the same policy to provide differentiated classes of delay service. TCP/IP data and UDP voice are separated into service queues, where the voice queue has the higher priority. We expect that reducing the mean queuing delay of one class will result in an increase in mean queuing delay for the other traffic class.

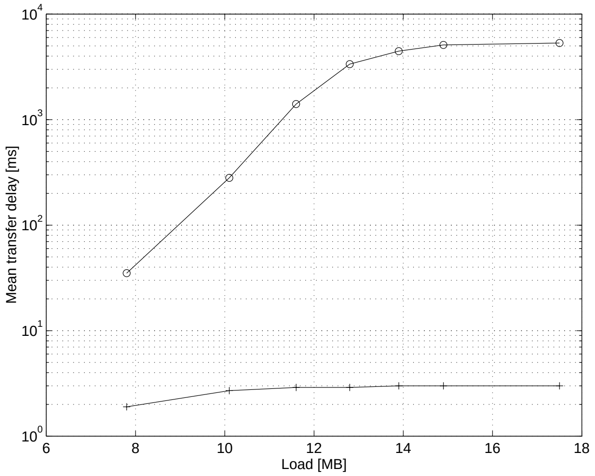


Figure 36: Two Service Queues for Internet Voice (lower line) and Data (upper line)

Figure 36 shows that adding one more queue improves the service quality that can be provided for Internet voice. The increase in TCP data does not significantly impact the voice service any more. The TCP data is still usable up to 10.5 Mbits/s total load, which is practically the same load as in the previous case. To summarise, adding the second queue has increased the service differentiation capability.

13.3.2 Scheduler

Wireless environment puts a special stress on the performance of the scheduling algorithm. This environment requires a scheduling algorithm that is efficient and aware of the QoS and traffic characteristics of the connections.

The scheduling algorithm has an important role in controlling the flow of the packets over the band-limited wireless channel. Used together with Call Admission Control (CAC) and resource allocation, scheduling can be used to guarantee the satisfaction of different QoS requirements for a different traffic types. Admission Control and resource allocation operate at the time of the connection is established, deciding whether new radio flows/connections can access to the channel. The scheduler decides which packets should be placed in each MAC frame. The scheduling algorithm should aim to provide the following properties: [Gar96]

- Maintenance of traffic characteristics of the connections
- QoS requirements satisfaction - the QoS parameters related to delay and loss are important to maintain according to the traffic contract.
- Statistical multiplexing gain - the scheduling should smooth or take into account the effect connections with variable bit rate traffic characteristics have on buffer occupancy (congestion).
- Utilisation of bandwidth unallocated or allocated to idle connections - since the applications (WWW-browser for instance) may not be sending packets all the time. The unallocated resources should be utilised during these silent periods.
- Declared and real traffic consistency - in the case where the source is producing more traffic than expected and thus breaking the traffic contract, the scheduler should for instance ‘drop’ the priority of the connection.

With the queuing scheme proposed in this document, the Scheduler could work for instance in the following way:

Firstly, the Scheduler prioritises the packets according to the three priority classes. Class 1 has highest priority and class 3 has the lowest priority. The scheduler begins to allocate packets pending in the class 1 queues. Inside the priority class, prioritising between packets/flows can be made according to delay requirements i.e. choosing the packet that has the least ‘lifetime’ left and so forth. Parallel to this the flows consuming less bandwidth than allocated have higher priority. This can be taken into account by using traffic policing function such as Token Bucket. If there is still space in the MAC frame when all of the priority class 1 packets have been allocated, the Scheduler begins to allocate packets from priority class 2 queues in the same way as in the class 1 case. After all the class 2 packets have been allocated, the Scheduler allocates class 3 packets into the free slots. The class 3 queue operates in a FIFO (First In, First Out) fashion. The scheduling ends when all of the packets have been allocated or when the MAC frame is full.

14. MOBILITY

14.1 Terminal Registration and Authentication

The terminal performs the IP level registration process when it has been powered on. It is also the initial part of an IP level handover. The process is the same in the terminal’s home network and in foreign networks – see the corresponding MSCs in Annex A.

The process is performed after the link level registration process. The link level procedures have already authenticated the terminal and accepted its access to the network. The link level entities have also made the terminal a member of the *all-nodes* and *solicited-node* multicast groups. The solicited-node multicast address is calculated from the EUI-64 identifier provided with the link level registration messages.

These multicast groups are local to the link to which the terminal is currently attached. This means that both the terminal and the M-router know that this terminal belongs to these multicast groups. The M-router knows to route packets that are addressed to these groups to this terminal and the terminal is able to receive and process these packets. This is why the terminal does not perform an explicit join to these groups using the link level multicast membership protocol in the beginning of the IP level registration process after it has generated its link-local IP address.

When the link level has performed its registration procedures the link level informs the upper level entities of its readiness. If this is a power on situation the network interface in the terminal moves into an enabled state. In this case the terminal generates its *link-local* IP address from the information provided by the link level (an EUI-64 formatted MAC identifier of the interface). Normally a host would validate this address by performing duplicate address detection procedures before the final assignment of the address. However, the EUI-64 identifier has already been verified during the link level registration procedures so there is no need to carry out this verification again as the link-local IP address is generated from that same unique identifier. The terminal assigns the link-local IP address to the network interface.

The following actions are performed in both the power on situation and in the case of IP level hand over. The host would normally use link level mechanisms to join the all-nodes and the solicited-node multicast groups. For the reasons mentioned above this task is not performed in this environment at this point since the task has been carried out implicitly during the link level registration procedure.

Once the network interface has been assigned a valid link-local IP address the terminal performs Router Discovery actions in order to find its default router and possibly to obtain the network prefixes for its *site-local* and *global* IP addresses. The M-router answers the terminal’s solicitation by an advertisement and provides the terminal with information on the M-routers link level address as well as its IP address. The terminal updates its default route to point to this address. Note that the Router Discovery process is necessary even in the home network because the home network prefixes might have been renumbered.

The M-router may advertise the site-local and global IP address prefixes for this link in the advertisement. If this is the case the terminal generates its site-local and global IP addresses based on these prefixes. Again, the terminal does not need to verify the uniqueness of the addresses by duplicate address detection procedures since the network interface specific suffix of the addresses has already been proven to be unique during the link level registration process. Also, in this case the terminal will probably not need to join the solicited-node groups of these addresses as these groups are potentially the same as the solicited-node group for the link-local address of this network interface. The terminal assigns its site-local and global IP addresses to the interface.

The bits in the Router Advertisement may also be set so that the terminal is required to acquire its site-local and global IP addresses via a stateful configuration mechanism such as DHCPv6. In that case the terminal sends a solicitation to the all configuration servers group on the link in order to find a server that is willing to serve the terminal. In this environment the server would reside in the M-router. The server responses with an advertisement so that the terminal learns the IP address of the server. The terminal sends a configuration request to this address and receives the requested addresses in the servers reply. The configuration server would probably add the terminal into the solicited-node groups of these addresses on behalf of the terminal. Otherwise the terminal would have to initiate the generic link level and IP level multicast group membership procedures to join these groups. In any case the terminal must enable the receiving of the packets coming from the interface and addressed to these solicited-node groups. The addresses do not need to be verified by the duplicate address detection procedures, the configuration server and the other entities within the M-router are supposed to know which addresses are valid for the terminal’s interface. The terminal assigns its site-local and global IP addresses to the interface.

The stateful configuration mechanism can be optimised further by leaving out the DHCP Solicit and the DHCP Advertise messages. They are not needed if we assume that the stateful configuration server is located in the M-router, whose address we know already from the Router Discovery process. This would be a non-standard deviation from the stateful configuration mechanisms and may not go in hand with the proposed IP level authentication methods.

Authentication of terminals is supported by the IPsec architecture’s Authentication Headers (AH). Usually, checking AH is an end-to-end operation. For address allocation and authentication, a server is needed. It is either built into the address auto-configuration process or hosted by a DHCP server. The latter approach allows generally for more comfortable and more controllable operation and administration.

Furthermore, confidentiality is supported by IPv6 IPsec architecture. Encapsulated Security Payload options (ESP) allow for confidential delivery of data. Especially in foreign networks this can be an essential requirement. It is also possible to couple IPsec with mobile IP to turn on and off ESP dependent on operation mode and location (e.g. own company, subsidiary abroad, or partner company).

14.2 Addressing Scheme

Configuring addresses in a mobile (wireless) IP scenario is not very different from the normal workstation interface when it is enabled or re-enabled in an IPv6 subnetwork. The basic idea is to form new addresses by combining the EUI-64 identifer (48 bits) with the subnetwork’s prefix (80 bits) to form a valid 128 bit IP address. The protocol used to perform this task is supported by new addressing modes, such as the all-routers multicast address, and IP options that encapsulate L2 addresses. This procedure is also known as Stateless Address Auto-configuration. The mechanisms used, however, are Router Solicitations (RS) and Router Advertisements (RA) which are described in RFC 1970 [Nar96] (Neighbour Discovery (ND) for IP Version 6) and in a more recent I-D [Tho98]. This document (the RFC) will be changed/updated soon. Developments such as Mobile IPv6 may have an impact on ND but the main changes announced at the LA IETF meeting (April 1998) will be a split in functionality. Since many features were initially included in ND, a split was proposed to separate essential and fancy features. The features needed for the WAND wireless IP system are on the essential list.

While Care-of Addresses (coa) are distributed via Mobile IPv6 binding update options making MTs reachable by CNs (correspondent nodes), it is sometimes desirable to use a different coa or to perform security checks prior to allowing access to a local wireless LAN. One way to restrict access to the wireless LAN is to use DHCPv6 (Dynamic Host Configuration Protocol Version 6) [Bou98]. This protocol assumes a working IP communication which can be achieved by performing stateless address auto-configuration or, more attractively, by using link-local addresses. DHCPv6 is prepared to deal with link-local scope addresses by providing a relay mechanism to forward configuration requests to a server. Address configuration via DHCP is also termed stateful address auto-configuration.

Since both protocols have their merits and are complementary rather than overlapping, this design note describes both protocols in greater detail in sections 3 and 4, respectively. This section concludes by listing the pros and cons of each protocol.

14.2.1 Stateless Address Auto-configuration

The simplest method to obtain new addresses is via Stateless Address Auto-configuration. IPv6 addresses are longer than typical L2 addresses (802.x for LANs, such as Ethernet or Token Ring, or E.164 addresses on ISDN systems). This makes it possible to find new addresses by prefixing the L2 addresses with the local site prefix. In IPv4 this was not possible and a previously known mapping from L2 address to IP address had to be performed.

In IPv6 there is also the notion of plug and play network configuration. This means that ad hoc networks can be set-up with no administrative overhead in link-local scope networks. Also, the addressing of appliances (phones, PDAs, games, coffee makers, etc.) becomes feasible [Hin98].

Table 10: Stateless Address Autoconfiguration.

MT	**Mobile Router (or any other host for duplicate address detection)**
In order to get RAs quickly the MT sends a Router Solicitation message to the *all-routers multicast address [FF02::2].* The source address is set to the unspecified address [0::0], indicating that there is no address configured for the MT. The home address should not be used here. Furthermore, the MT should insert its L2 address as an IP link-layer address option. An MT can retry (3 times) every second if there is no answer.	A mobile router (MR) can send unsolicited Router Advertisements (RA). However, they must be delayed by at least 3 seconds which means there is an unacceptable waiting time for MTs.
	The MR sends back a RA which is specific to the solicitation (request) message (addressed by using the L2-option). The MT requires the link-prefix (subnetwork-prefix) and the MR address to send packets to unicast addresses not residing on this network.
The MT combines the link-prefix with an EUI-64 identifier to form a global IPv6 address.	
There may be duplicates on the subnetwork. Therefore, a DAD (*Duplicate Address Detection*) step is performed.	
This is performed by sending *Neighbour Solicitations* (NS) to the solicited node multicast address (which was also joined by the host which is checking the address). Source address is unspecified. Since NS must be performed according to the specs [Tho98] and the normal case is that nobody answers, this usually takes a long time - 3 retries, with a second delay between each plus a random initial delay.	Neighbours receiving a solicitation, check for validity and the target in the message. If the target is the same address, a *Neighbour Advertisement* (NA) is sent back to the soliciting host telling it that the address is already taken.
If a NA is received for the configured address, the address can not be used and the interface must be disabled.	
If no answer is received, the address can be configured and used. Furthermore, the previously obtained RA can be used as the default router.	

The steps to be performed to obtain an address include Router Solicitation and Neighbour Solicitation. Router solicitation is used to learn the link-prefix and default router address (in our case the mobile router) and Neighbour Solicitation is performed to check for duplicate addresses. The protocol details are given in Table 10.

Using this method for the WAND wireless IP scenario, no changes to the specifications are needed for proper operation given IP multicast can be used. WAND-L2 addressing can be used for the RS/RA as well as NS/NA protocol. However, the DAD mechanism relies on sending Neighbour Solicitations and Advertisements to the *solicited-node multicast address* of the address being configured. Thus the wireless IP system must support this.

Furthermore, the DAD processing can take a long time. The basic problem seems to be the fully distributed approach in which all hosts participate.

14.2.2 DHCPv6

The Dynamic Host Configuration Protocol for IPv6 pursues a different, more controlled approach to address configuration. A configuration server (as known from DHCPv4) is used to assign addresses upon receipt of L2 addresses. In IPv4 this was the only alternative to manual configuration and some boot-protocols. DHCP provides much more than address configuration, it was built as an initialisation server for hosts. This means that can serve any configuration parameter useful to administrate hosts on an internetwork.

Just to note one of the many design goals of DHCPv6, it is stated in [Bou98] that the protocol must be compatible and inter-operable with Stateless Address Auto-configuration. The same applies to manually configured hosts.

DHCP defines *clients* (hosts to be configured) *servers* and *relays*. Servers provide configuration information and are maintained by system management. Relays are used to forward information to clients and servers when direct communication is not possible (e.g., when using link-local addresses). The basic messages defined by DHCP are given in Table 11.

Table 11: DHCPv6 Messages

Solicit	Multicast message from client to servers or relays.
Advertise	Unicast message sent back to the requesting client upon receipt of a solicitation.
Request	Unicast message from client to server to configure a specific parameter.
Reply	Unicast message from server (or relay) to client containing the parameter/value pair.
Release	Unicast message. Client is indicating the release of local resources (e.g. an address that is no longer used).
Reconfigure	A unicast or multicast message from servers to clients indicating a change in previously configured parameters.

For address configuration the bootstrap process starting with DHCP solicitations is of most interest. A solicitation is sent to one of three well-known MC addresses (all-DHCP-agents, all-DHCP-servers, or all-DHCP-relays). Furthermore, datagram communication is fixed on two well-known ports (to client: 546, to server: 547). Firewalls need to permit these ports.

Table 12: DHCP Messages for Stateful Address Configuration

MT	**Mobile Router / DHCP relay or server**
First, a link-local address is formed by prepending the well-known link-local prefix [FE80::0] to the EUI-64 interface identifer. The prefix FE8 is 10 bits which restricts interface identifiers to be less or equal than 118 bits. DAD must then be performed to verify the interface identifier's uniqueness on that link [Tho98].	
This address is used as the source address when communicating with a DHCP server or relay. The client starts by sending a DHCP solicitation message to all-DHCP-agents (this MC address includes servers as well as relays).	
	A server can answer the solicitation directly with a DHCP server advertisement message.
	A relay forwards the solicitation to servers and relays the advertisement back to the requesting clients link-local address.
Upon receipt of a server advertisement a request for the address is made. In [Bou98] the semantics of the DHCP configuration parameters are not described. It is not specified how the MT is identified or authenticated. A simple version could use the link-local address, a more complex one should use a form of digital signature by the MTs owner.	
	The server sends back a DHCP reply (or via relay) with the configured address.
MTs can release the assigned addresses if necessary.	

The basic procedure starts by using link-local addresses with the well-known (i.e. reserved) address prefix [FE8::0]. Using the EUI-64 interface-identifier a temporary address can be formed. This is used as the source address for DHCP solicitation.

To process such a solicitation, a server or relay must be present on the local link. Relays perform forwarding in both directions. Servers can reply immediately with a DHCP advertisement message. The detailed protocol processing is given in Table 12.
Stateless and stateful address auto-configuration can be used to assign care-of addresses to mobile terminals when they are operated in a IP subnetwork.

Stateless address auto-configuration is a simple method that relies on link-prefix and L2-addresses to form global IPv6 addresses. The method to obtain them requires the standard Router and Neighbour solicitation and advertisement mechanisms. The latter is used for duplicate address detection and requires multicast capabilities.

To implement DAD correctly, all hosts on a link that have joined the solicited node MC group address must be informed of the solicitation packet.

For DHCPv6 a similar mechanism based on the well-known link-local prefix is used. Again, DAD must be performed. Furthermore, multicast addressing is used when looking for DHCP agents. This form of MC addressing could be avoided in the WAND system since the M-router usually implements the DHCP relay or server.

If MC can be properly implemented, DAD is still a bottleneck since it is based on long-lasting time-out mechanisms. A solution to this problem, however, is the inversion of the DAD protocol from a negative (“if nobody answers within 4 seconds you can take the address”) to a positive behaviour implemented centrally on the M-router. The M-router must record all address assignments since power-on of all of its APs and keep them in a suitable data structure. By doing this it can answer a neighbour solicitation with the unspecified source address (that’s DAD) immediately with a positive indication when the address is not present within the subnet. This is a minor extension of the DAD protocol necessary because the time-out approach is not suitable for handover and registration. Furthermore, ND RFCs are being reworked at the moment, so it may be possible to change the specifications.

To summarise IP level addresses are constructed based on the standard address configuration methods. The network interface specific part of the IP address is formed from the link level MAC identifier of that interface. The identifier is in the EUI-64 format. This identifier is verified during the link level registration process and is proven to be unique among all the terminals that are attached to the same subnet. If the verification fails the terminal is not able to use the network.

The link-local IP address is generated from a static well known prefix and the network interface identifier. The link-local address is not verified by the Duplicate Address Detection process, since the M-router is able to check for duplicate addresses.

The network interface’s site-local IP address is formed from the site-local prefix received in the Router Advertisement and from the network interface identifier. The site-local prefix uniquely identifies this subnet within the site area. The site-local address may also be acquired from the stateful configuration server. In both cases the generated or received address is assumed to be valid and no Duplicate Address Detection is performed.

The network interface's global IP address is either generated from the global prefix provided by the Router Discovery process and from the network interface identifier. The global address may also be received from the stateful configuration server. The global address is not verified by the Duplicate Address Detection process.

The information in the M-router that is used in the terminals' address allocation process (either stateless or stateful) is managed by methods that are outside the scope of this document. It is also assumed that the terminals are attached to just one link at a time so that only one M-router is accessible at a time. The addressing scheme deviates from the standard procedures by the fact that no Duplicate Address Detection is performed. Instead the M-router keeps track of the IP addresses already in use on that subnetwork.

14.2.3 Security Issues in Address Allocation

Terminal/user authentication and data encryption are the two most considered security issues in the wireless networks. The data encryption here concerns both wireless link encryption and IP encryption. However, data encryption is a trivial problem that is not wireless specific and is hence out of the scope of this document. While a mobile terminal (MT) roams to a foreign network, they have to authenticate each other to establish the authenticity of the exchanged addressing messages due to the following security issues [Gud98]:

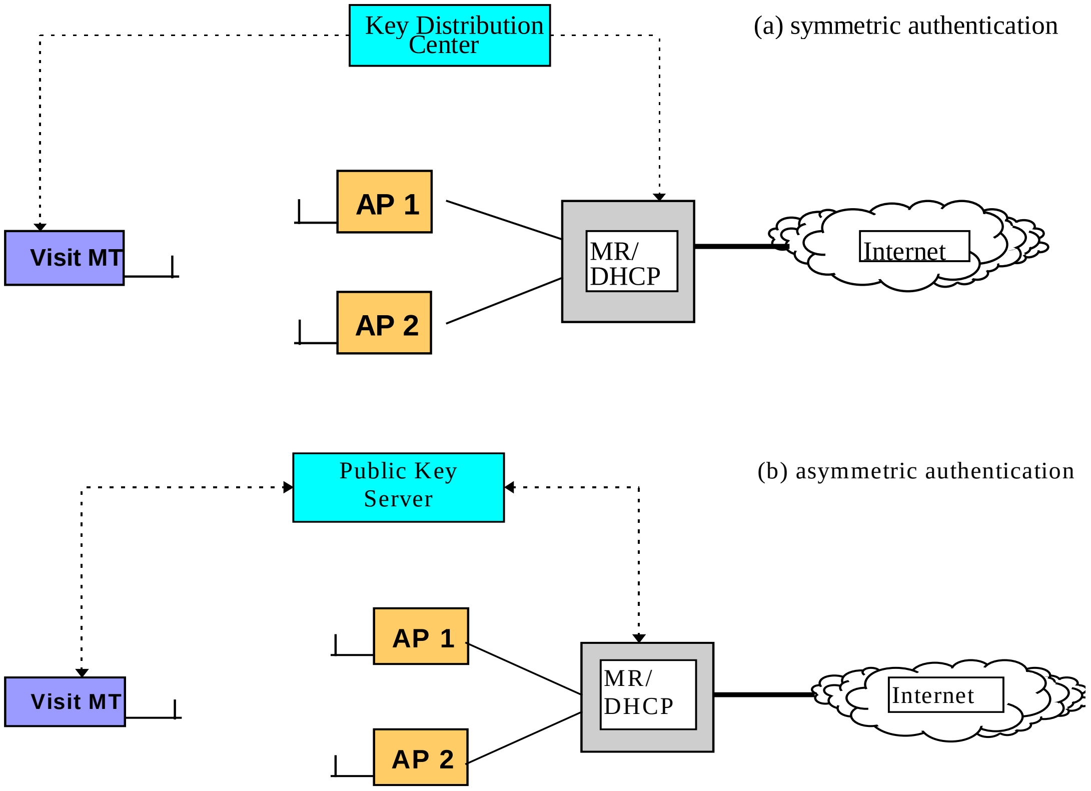
- DHCP Request messages sent by a mobile terminal masquerading as an authorised host are honoured without cryptographically verifying the authenticity of such messages.
- An adversary, which was refused allocation of an address, may listen to the addressing messages and configure itself as another granted host, which is referred to as a replay attack.
- Fake servers can provide clients with partially correct information that allows the attacker to route traffic through certain hosts where critical information can be collected, or configure clients with addresses of other clients which cause address conflicts.
- Servers can be attacked through exhausting the servers allocated address space or overloading the servers causing them not to respond to clients, which are called Denial-of-Service attacks.

In IPv6 stateless address auto-configuration, all hosts on the subnetwork are involved in the address allocation process, and therefore authentication between MT and all hosts is required. When using symmetric authentication methods, shared secrets need to be distributed between the MT and each host on the subnetwork. When using asymmetric authentication methods, all hosts should have the capability to process asymmetric authentication which requires the deployment of public key infrastructure and more computation power than symmetric authentication. Thus it isn't practical to apply authentication in stateless address configuration in terms of the complexity and the requirements.

Making use of DHCPv6 for address allocation, authentication is only necessary between DHCP servers and the mobile terminal. Either symmetric or asymmetric methods can be used for this purpose. Once the security association is established, the IP authentication header [Atk95] can be used to perform the data origin and integrity authentication. DHCP servers can work as centralised access control and authentication servers. This design only focuses on authentication in the stateful address configuration procedure.

In the case of symmetric authentication (as shown in Figure 37a), a MT is introduced to a foreign network via local administration. Then the key distribution centre is requested to distribute the shared secret to both the MT and the corresponding DHCP server through a secure channel. Based on the distributed secret, the MT and DHCP server can authenticate the following exchanged addressing information between them.

Using asymmetric authentication methods, as shown in Figure 37b, the MT and the responsible DHCP server need to fetch the public keys of the communicating partner from a public key server, which has the certification authority (CA). They are suggested to keep these keys for the duration of the MT's visit in the foreign network to facilitate future message authentication. DHCP servers can also be configured with an access control list to check the access rights of a foreign terminal. To implement authentication in a global context, e.g. between companies, a hierarchy of CAs or PGP Web of Trust certificates can be used.


(a) symmetric authentication

(b) asymmetric authentication

Figure 37: Authentication in DHCPv6 address configuration

In order to prevent replay attacks, addressing information communicated between servers and clients should be encrypted.

In the case of a MT moving to a public network like an airport mobile network, it is not possible to use symmetric methods for authentication purpose due to the lack of the trusted introduction. But with the help of the available public key infrastructure, authentication can be achieved through asymmetric approaches.

In the real system, a hybrid authentication mode [Lit98] can be used. That is, the servers use public-key techniques to authenticate themselves to the users, and establish an encrypted connection. Then the user can use the legacy authentication system, like challenge response techniques, to authenticate itself to the servers. The assumption in this mode is that deploying public key for a small number of entities (DHCP servers) is possible without a full-blown public key infrastructure deployment.

14.2.4 Address Transparency in Mobile IP

This section describes the problem of addressing a mobile terminal. As stated in [Joh98] the MT can always be addressed by its home address. It is also possible that the MT uses the home address when sending (home address option, HAO). However, it is important to keep in mind that (a) sometimes one would not want to address the MT by its home address for efficiency reasons and (b) the MT does not always wants to use a HAO (18 bytes additional header per packet). Furthermore, option processing is not cheap, but possibly not a big problem at wireless speeds.

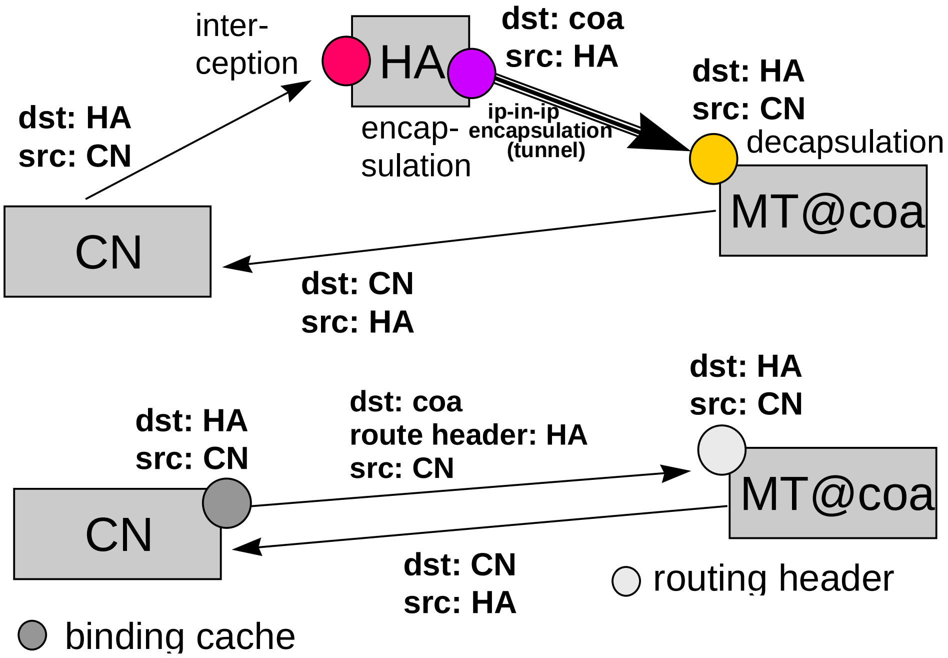


Figure 38: Addresses that can be employed with Mobile IP

The figure above shows the possible addresses that are used as source and destination addresses and that are optionally available through ip-in-ip headers (tunnels), routing headers, and Mobile IP options.

Some observations:

- The home address can be made available but it needs to be option-processed or ip-decapsulated.
- Packets addressed by the CN directly to the MT do not contain home addresses but could be grabbed from the routing header (when the MT is addressed directly, CN is using its coa but adds a routing header, containing the home address which is inspected upon arrival at its destination).

Now back to the question of how to identify the MT. More precisely we need to ask how can traffic to and from the MT be identified? Basically there are two possibilities:

- Always use the home address as the identifier. This is possible for traffic coming from the MT when the HA option is used. For traffic to the MT this is only possible when it is routed via the HA. Directly addressed traffic does not contain a home address (or must be obtained by processing routing headers which was not the intention).
- Use the current care-of address as ID. This is always possible for traffic from the MT without using any options. For traffic to the MT this is even possible when triangle routing is used due to the ip-in-ip encapsulation. But how can the mobile router associate the care-of and home addresses? This state needs to be kept, either in the binding cache or in another data structure (because the coa can change). It must be updated on hand-over for terminals entering and leaving the wireless IP subnet.

The choice between the two proposals depends on the need to identify inbound traffic. If this is not a critical issue the burden of keeping state or sending HA option should be traded against each other.

14.3 Intra IP Subnetwork Mobility

When the MT roams within the area controlled by one M-router (intra-AP and intra-subnet handovers), it remains within the same IP subnet, and thus retains its assigned IP address. The hand-over involves the switching of the control (MM, RSVP and IFMP) and data connections (BE, QoS) from the previous to the new AP which can be realised through low level VC diversions as discussed in [Kal98]. CAC (Connection Admission Control) has to be performed for the new target AP. There are two CAC functions, one for the fixed network (Fixed CAC - FCAC - which refers to resources in the switch - AP link), and one for the wireless part (Wireless CAC - WCAC - which refers to radio resources). FCAC can be handled by standard IP level CAC mechanisms (e.g. RSVP’s traffic control entities), while the WCAC is performed in a centralised way within the M-router, on the basis of information provided by the involved AP. The QoS characteristics of each active connection are maintained in the M-router (possibly within the Flow Manager). If CAC algorithms grant admission, VCs pertaining to the moving MT are diverted to the new link (by means of GSMP or some proprietary switch API). If the new AP can support lower QoS than requested, the MT has the option of:

- accepting the modified QoS offered by the network,
- converting the connections to best-effort, or

- completely dropping the connections.

In such cases, an indication of the change of QoS must be propagated to the correspondent node. For example RSVP messages indicating modifications or cancellations of the reservations are propagated to the correspondent node. Resources allocated to the connections in the old/source AP are released through the use of explicit signalling (instead of leaving the reservations to time-out). The primary goal during the execution of a handover is to preserve the control channels (RSVP, MM and IFMP VCs) in the new AP, and if possible the data channels. Since the handover is confined within the same switch, no Mobile-IP specific signalling is exchanged.
MM signalling is exchanged though, to notify fixed network entities of the hand-over and force the reconfiguration of the access network. The signalling required for the intra-subnet scenario is quite similar to the one presented in [Kal98] for the present form of the WAND pilot.

A similar approach is suggested in [Ach98] where a two level mobility management scheme is proposed. Mobile-IP is used to provide mobility support within the inter-network as a whole. In the same paper the concept of "mobile ATM" is introduced. Within a mobile ATM cloud (not necessarily comprising one ATM switch), location management and handovers are handled by proprietary ATM signalling. Mobility of a terminal within the ATM cloud is transparent to Mobile-IP. When the mobile node initially enters the mobile ATM network, it is assigned a home ATM address and a foreign IP address. As required by Mobile-IP, the terminal's foreign address is communicated to its home agent located somewhere in an interconnected IP cloud (possibly another ATM cloud). Subsequent moves within the mobile ATM network generate location updates only at the mobile ATM level. Each move causes the terminal to be assigned a new foreign ATM address while the terminal's home switch (the radio port to which the terminal has initially attached) maintains a mapping between the home and foreign ATM addresses. The mapping between the foreign IP address and the home ATM address never changes as long as the MT remains within the mobile ATM cloud.

14.4 Inter IP Subnetwork Mobility

When an MT moves to an AP connected to a different router than that of the previous AP, an inter-IP subnet handover must be performed. A new coa is obtained by the MT in the new subnet, in addition to the establishment of control channels (IFMP, RSVP, MM). Address allocation is performed by the Stateless Address Auto-configuration mechanism, which is based on a reserved link-local subnet address and the interface identifier. Using this temporary address, the MT can either obtain a local subnet address using the Neighbour Discovery protocol or contact a DHCP server to obtain an administered address. The newly acquired coa is registered with the MT's Home Agent by means of the Binding Update and Binding Update Acknowledgement messages. The MT also sends Binding Update messages to each CN and the previous M-router. Thus, any packet received by the previous M-router will, for the relatively small duration of the handover, be forwarded to the new sub-network (using IP-in-IP encapsulation) (see Figure 39). This is aligned with the objective of keeping handover lossless.

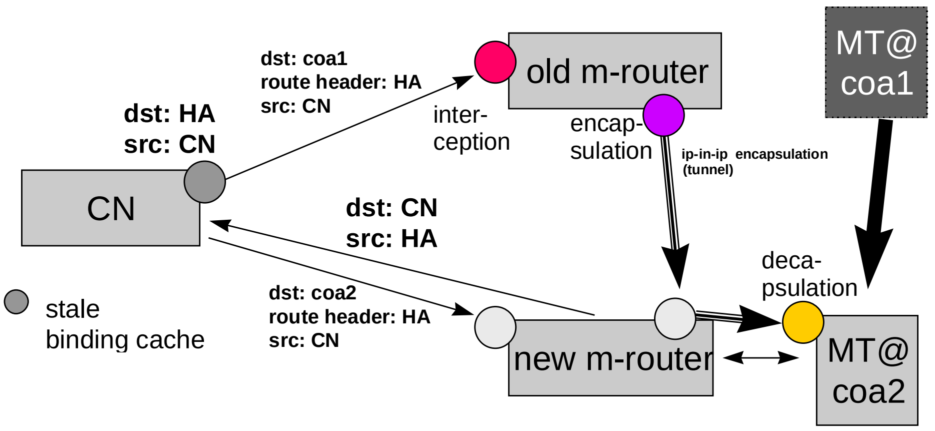


Figure 39: Forwarding packets between old and new m-routers through ip-in-ip encapsulation

The preservation of QoS guarantees during inter-subnet hand-over depends on the IP level QoS mechanism employed. QoS guarantees can be maintained within the wireless IP subnets by the MT informing the new M-router of the QoS requirements of each active flow. However the maintenance of QoS support within the backbone Internet will depend on whether diffserv, RSVP or another QoS approach is employed.

When diffserv is employed at the IP level, each packet contains a DS field describing the resource requirements for the packet. Thus when the MT obtains a new coa, this will not impact the QoS provision across the core network.

In contrast, in RSVP if the MT moves, the path to the CN will have changed, hence the RSVP path needs to be updated. However, the address of the MT has changed as well. This means that the new data packets will look like they are for a new flow from the router’s perspective. Hence a new reservation is needed end-to-end. However, the RSVP packet itself will contain the MTs home address. Thus when the RSVP module in routers and the CN process the packet and reserve resources this will be on the basis of the MTs home address not the coa. Hence there is a mismatch between the outer-header information and the inner RSVP information.

An RSVP session is identified by the triple: destination address, protocol id and optionally destination port. A filter spec along with the session spec defines the set of packets to receive the QoS defined in a flow spec. In IPv6 the filter spec is often given by the source address and flow label. In the context of this DN, it is important to note that the session and filter specs will contain the MT home address, rather than the coa, since the RSVP module lies above the IPv6 module.

The other complication is that RSVP PATH and RESV messages carry a RSVP_HOP information element. This is updated every hop and contains the unicast IP address of the outgoing interface. PATH messages are carried with the same source and destination IP addresses as the data. This means they will be routed correctly through non-RSVP clouds (note they are processed at each hop and the RSVP-HOP field modified). In contrast RESV messages are sent hop-by-hop, where the destination address is the unicast address of the previous RSVP hop. The implication on this for MTs, is that when the MT is the receiver and is thus sending

RESV messages, the routers are the destinations of the RESV datagrams. Hence the home address option will be processed, and thus the RSVP module will always see the home address of the MT, rather than the coa. Moreover, the Mobile IP binding update message travels end-to-end so the routers along the path will not have a binding cache with the MTs coa and home address. If the MT is a sender it will be transmitting PATH messages. Given PATH messages travel end-to-end, the intermediate routers will process the packets with the MT's coa as the source address, even though the inner filter spec address will be the MT's home address. Hence there is a significant mis-match between the IP addressing viewed by routers along the path between the MT and CN.

PATH messages are sent end-to-end, hence when the CN transmits a PATH message it will contain the MT's home address, the IPv6 module must then swap the destination address for the MTs coa and forward the message to the receiver. The CN must also enter its IP address as the pervious hop entry. When the PATH message reaches the first router, it must notice that the datagram contains an RSVP message (via the protocol id) and pass the PATH message to the RSVP module for processing. The RSVP module creates PATH state and must forward the PATH message on the basis of routing information, it retrieves from the routing module. The routing information is based upon the session destination address and optionally also the sender's IP address and includes a list of the IP addresses of the interfaces to which the PATH message should be forwarded. The RSVP module includes this information in the previous hop field before forwarding it out the correct interface. The problem is that we do not want the PATH message to be forwarded to the MT's home address but directly to its coa. It is important to note that if the MT is the sender, there will be no routing information problem, as long as routing decisions are based upon the destination address only and not the source address. If the MT is a sender, the PATH messages will be routed correctly to the CN. Since the RESV messages are forwarded on the basis of the previous hop information they will be routed directly to the MT's coa, even though the routers do not contain a binding cache for the MT.

This leads to the more general question of how the routing table and binding cache are inter-related - i.e. if RSVP looks up the routing table for the next hop to the MT on the basis of the home address will it get the correct information for the coa**.** We assume that if a binding cache is present when RSVP consults the routing table the home address is mapped to the coa, so the RSVP module gets routing information based on the coa. From an investigation of the CMU IPv6 code [CMU97] this appears to be the case.

The above process is possible at the CN and MT, however the intermediate routers will not have a binding cache for the MTs home and coa. Figure 40 shows the standard operation of MIPv6 and RSVP

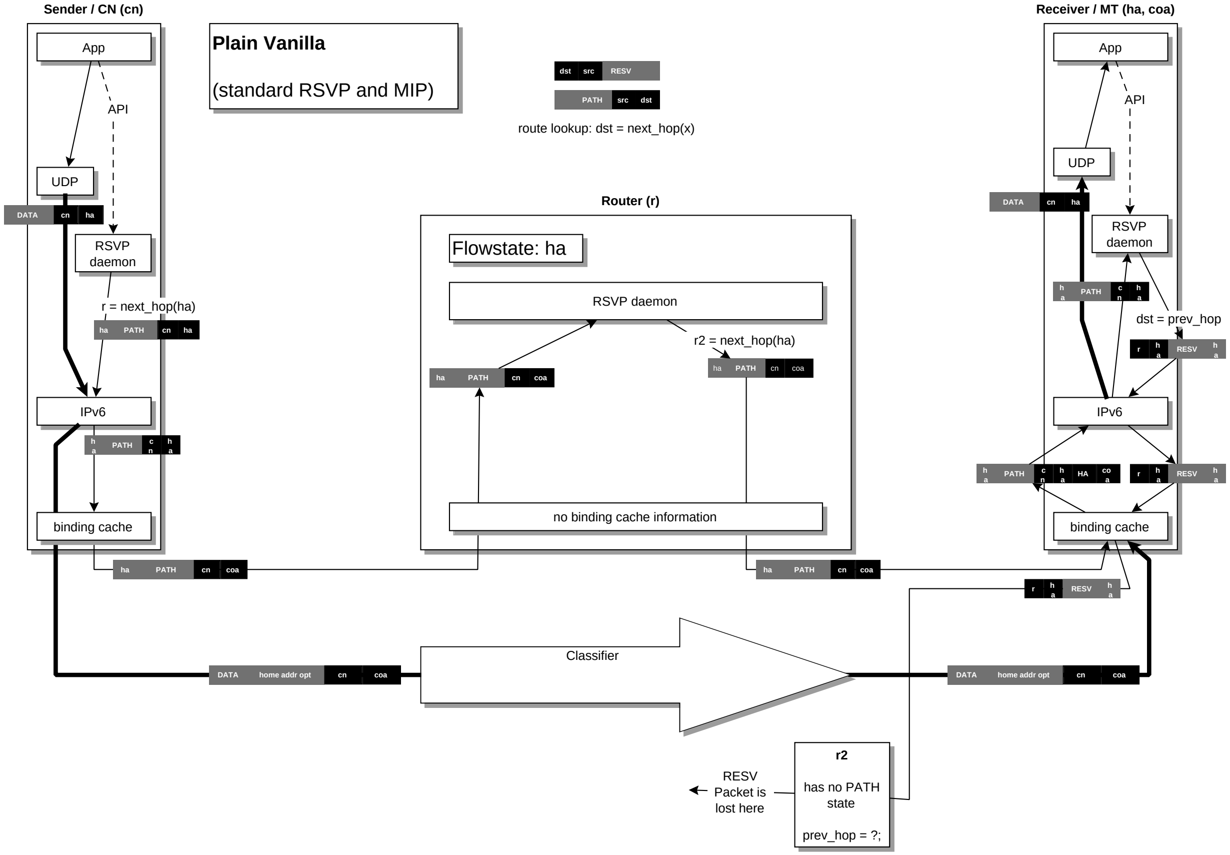


Figure 40: Standard MIPv6 and RSVP Operation

We assume a topology as shown in Figure 41. We focus on traffic flows from the CN to the MT since as mentioned above, routing problems do not occur for traffic flowing in the reverse direction. At the CN RSVP PATH messages are addressed to *next_hop(ha)* which will mean the packet is forwarded to the next hop router on the path to the MT's coa (see Figure 40). At router r the same lookup process occurs. Since the RSVP daemon (rsvpd) at the router is unaware of the MTs coa, it will set the previous hop field to be the address of the outgoing interface on the path to the HA. However it will forward the packet based on the coa address and hence towards the next hop router on the path to the coa. This means that flow state (information obtained from the RSVP messages to identify the packet stream and its QoS requirements) will be created at all routers on the path between the CN and the MTs coa, however the previous hop information at each of the routers will be incorrect. Thus when the MT transmits a RESV message it will not be routed along the direct path from the MT to the CN. Instead it will be routed, based on the incorrect previous hop information, to a router which has no flow state information because it did not receive the PATH message. Hence it is not possible to reserve resources between the CN and MT using MIPv6 and RSVP in their current forms.

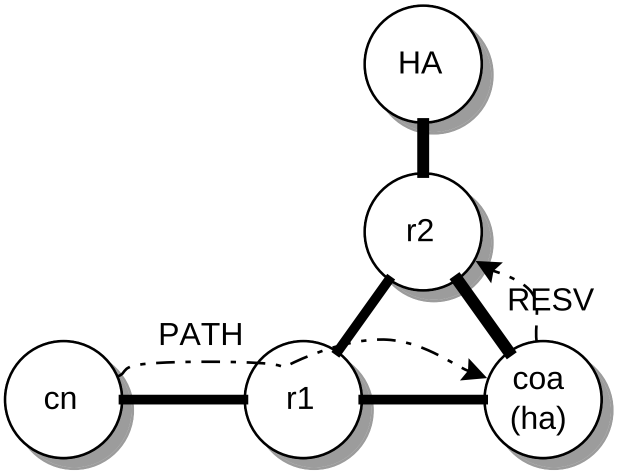


Figure 41: Standard MIPv6/RSVP Topology

The discussion to date has described the problem of routing PATH and RESV messages correctly between the MT and CN. Another problem that must be considered is how to provide the requested QoS on the new portion of the path when the MT gains a new coa in a timely fashion.

To summarise there is a mis-match between RSVP and Mobile IP. The problems that must be resolved are:

- How to route RSVP PATH and RESV messages directly to the MT coa.
- How to reserve the required QoS on the new path quickly when the MT moves.

14.4.1 Solving the Routing Problem

14.4.1.1 Change RSVP at CN to operate on the MTs coa

In this approach the RSVP module at the CN operates on the coa rather than the home address. IP addresses within RSVP PATH and RESV messages will be the coa, not the home address. The benefit of this approach is that no changes to intermediate routers are required because the RSVP message contains the correct information for the current path. The PATH messages will be routed correctly to the MT's coa. However it means changes to RSVP are needed, and the mobility of the MT is no longer transparent.

The coa of the MT could be obtained by consulting the binding cache in the CN. This does mean that either the RSVP module needs to be informed when the binding cache changes (i.e. updated when a binding cache update arrives), or, the RSVP module must check the binding cache every time it sends an RSVP message.

This approach follows the basic approach of setting up flows for the actual path, rather than for the MTs home address, i.e. the flows are set up based on the actual destination address information.

Two options are possible:

The first approach requires modifications to both the RSVP daemon (rsvpd) and MIP. In this case MIP must provide an interface so that the rsvpd can look up the next hop for the mobile address, i.e. the rsvpd translates the MT home address to the coa via the binding cache, before it fills in the PATH messages. Not only is the next hop selected based on the coa, but the coa is actually used as the destination address in the PATH message. This is shown in
Figure 42.

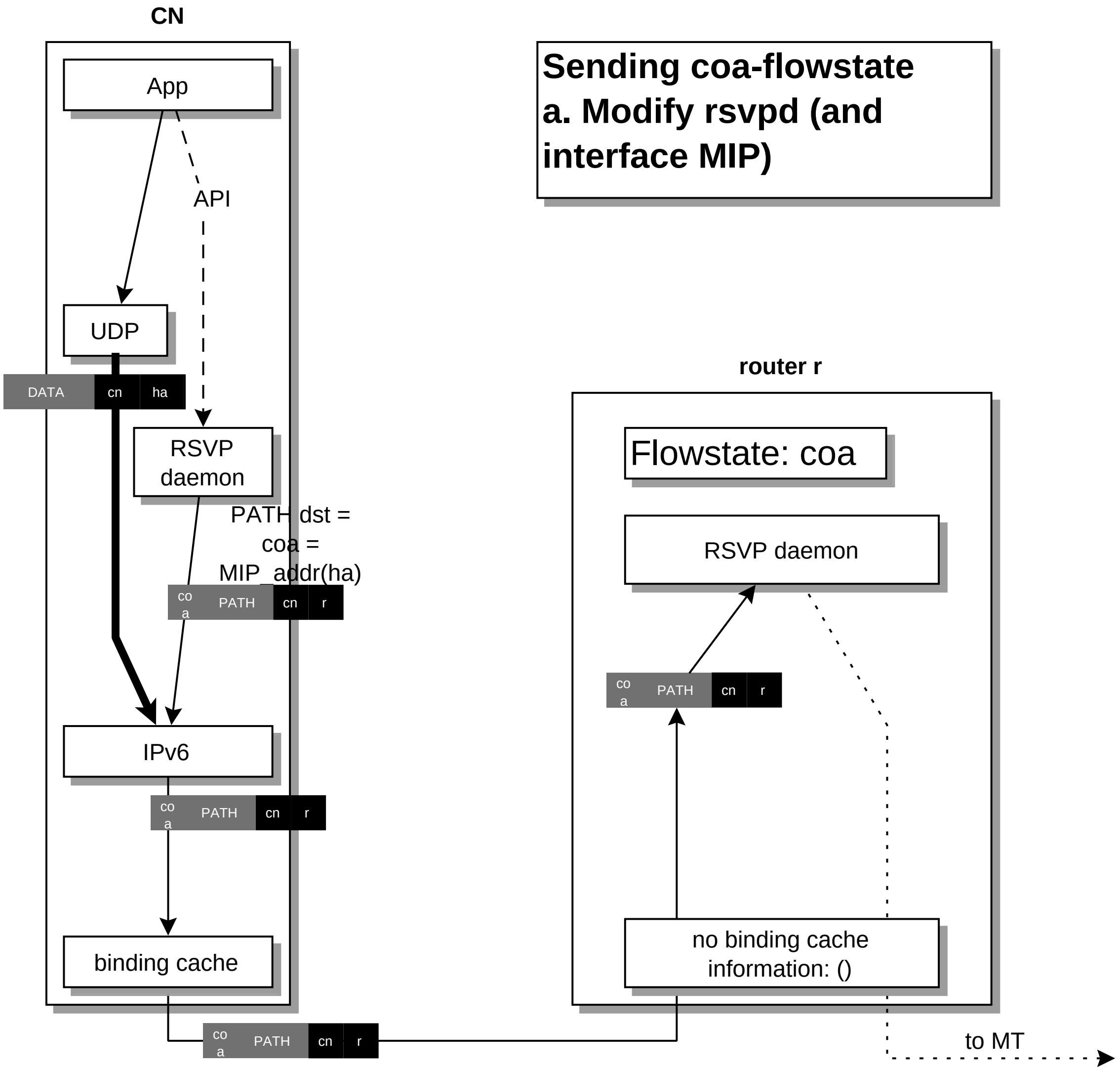


Figure 42: Base Flow State on coa and modify RSVP and MIP

An alternative is to modify MIP only. In this case RSVP implementations do not need to be modified. The binding cache is aware of RSVP messages and swaps the home address in the RSVP messages with the coa of the MT. This is shown in Figure 43.

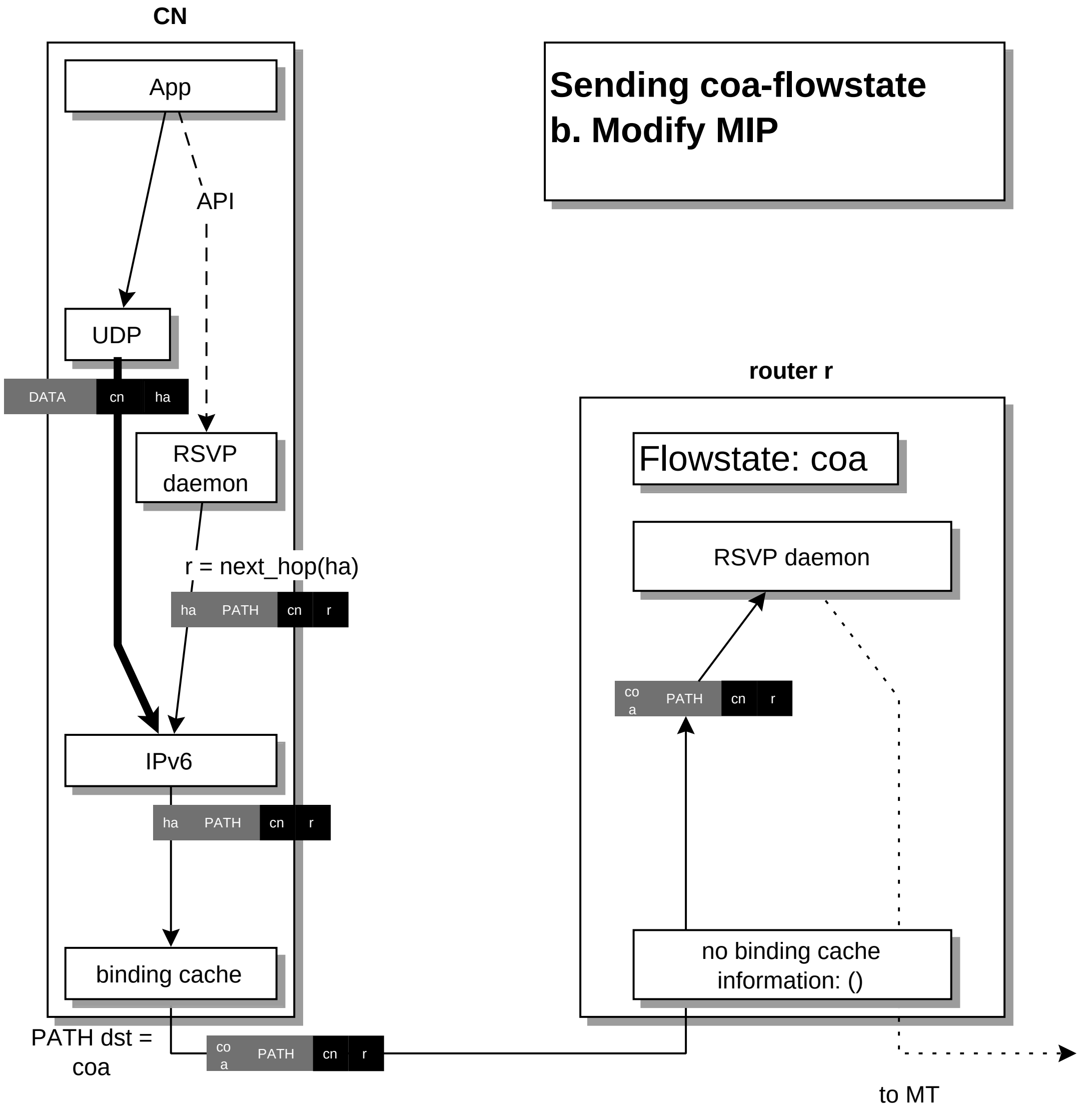


Figure 43: Base Flow State on coa and modify MIP only

The advantage of this second approach is that it does not require changes to RSVP implementations. However the first approach is a much cleaner solution, in that MIP does not need to understand RSVP specific information. There could also be possible performance problems with this second approach since all packets need to be checked for PATH messages. This is especially undesirable for hosts having high-speed interfaces.

A disadvantage of modifying flow state to be based on the coa is that the intermediate routers will assume this is a new RSVP reservation flow. Hence there may be situations where the reservation is denied, because the old reservation is still active (hasn’t timed out yet), but there aren’t sufficient resources to create the new reservation as well. This problem could be overcome if there was a new RSVP exchange message that contains the old and new reservation details. The intermediate routers could then update the RSVP state information (i.e. the MTs address from the contents of this message). The problem with this is that it requires changes to the RSVP specification. This is one reason why it may be better to modify RSVP capable router implementations to recognise that flows that differ only in the coa, are actually the same flow, rather than create a new message.

14.4.1.2 Modify packet filters and classifiers at RSVP capable routers

In this case router packet classifiers are modified so that they are aware of both the MT's home address from the RSVP module and the MTs coa. In this case the classifier maps all coa's to the same old RSVP state.

The issue is how do the classifiers and filters learn the MTs home address.
One option would be for the binding update option be made a hop-by-hop option for RSVP capable routers. This would allow routers to build a binding cache that contains the MT home address - coa mapping. This would have to be a MUST requirement. This is because if it was a SHOULD process hop-by-hop when the RSVP messages meet a router without a binding cache the packets will be re-routed to the Home Agent. A disadvantage of this approach is scalability, in terms of the size of binding cache that would need to be maintained, particularly in the case of backbone routers.

Here the routing problem (highlighted in Figure 40) is resolved by updating all binding caches on the path between the CN and MT. This prevents the routing of PATH messages to the HA as shown in Figure *44*.

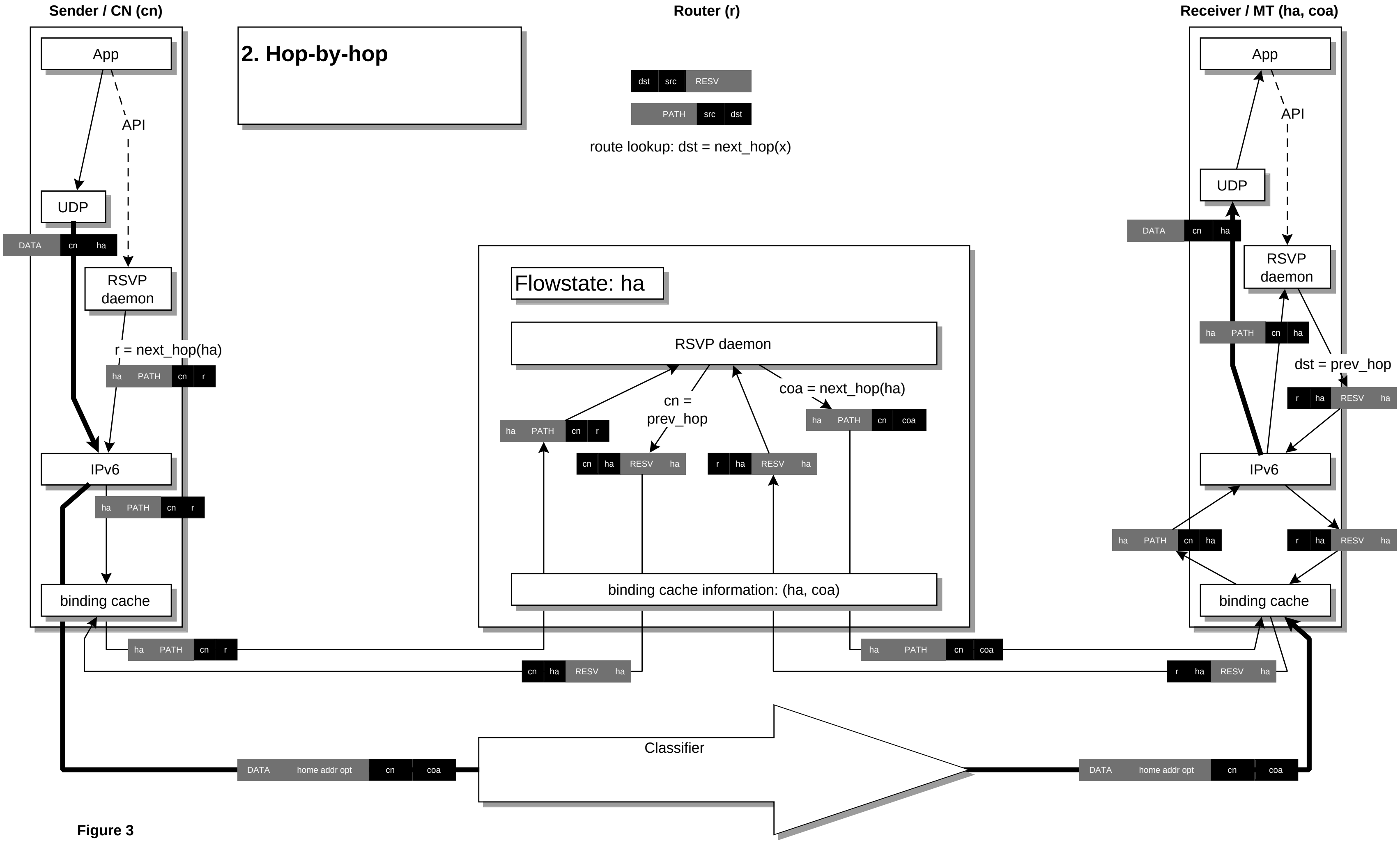


Figure 44: Create a Binding Cache Hop-by-Hop

The benefit of this approach is that it requires changes to Mobile IP but not to RSVP. Also there may be other protocols affected by the problem of the MTs coa
changing. Thus treating it at the Mobile IP layer will provide a more general solution. Furthermore, there are several RSVP implementations and RSVP is on the standards track, whereas Mobile IPv6 is not so stable yet.

Generally this approach works fine. However routing asymmetry can lead to the same routing problems. For instance consider multiple routes as shown in Figure 45. In this case binding cache updates (BCUs) travel via r2 and PATH messages travel via r1, that is the route between the MT and CN is different in each direction. Since r1 did not receive a BCU, packets will again be sent to the HA. Research has found that routing asymmetries are relatively common [Pax96].

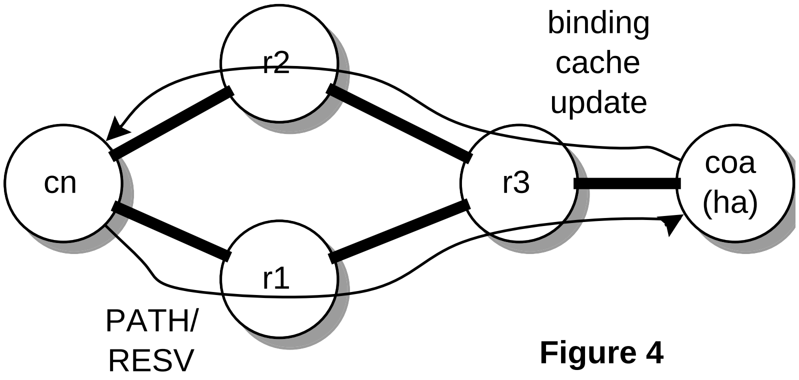


Figure 45: Asymmetric Routing Problem

Another approach would be to change the implementation of RSVP capable routers so that when PATH messages are received, even though the router is not the destination of the packet, the destination IPv6 headers (in particular the home address option) are processed. This would allow the router to learn the home address of the MT without having to make changes to the Mobile IP specification.

14.4.1.3 Modify RSVP capable routers to pass outer header IP addresses to RSVP

One of the key problems with the standard approach is that routers will make the reservation for a flow containing the MTs home address. Yet the router classifier will see data packets with the MTs coa address and will not realise they should receive the reserved resources. Rather than creating a per-router binding cache, an alternative is to pass the outer header IP addresses to the RSVP module. This would allow the PATH state to be created on the basis of the coa. This would mean the flow identifier would match the header fields in the data packets.

This is possible because packets containing PATH messages are addressed to the destination proper, not to the address of the next hop. With RESV messages this would not be possible because the packet containing the RESV message is addressed to the next hop, not the CN. Hence the RSVP module would need to maintain a mapping between the MT home address and coa. Given this, one could ask why not use the binding cache solution described in the previous section. The advantage of this approach is that it does not suffer from the route asymmetry problems because the mapping information is built up from RSVP messages rather than the MIP message flows.

The RSVP module will have the MT's home address and coa. This means that the other advantage of this approach is that when the MT moves again (causing the coa to change), the RSVP module in routers that are on both paths will know it is for the same flow. Thus they can change the reservation details, rather than having to tear down one reservation and create a new one.

In this case the session id is used for identification purposes only, and routing decisions should be based on the coa so that traffic is routed directly to the MT.

The disadvantage of this approach is that it requires changes to RSVP implementations for IPv6. However it does not appear that changes to the specification itself would be required.

14.4.1.4 Path Extension Approach

If the MT begins a RSVP session while in its home network the initial reservation will be made across the network correctly. However if the MT starts the session anywhere else one of the approaches described above must be employed to reserve the resources between the CN and the coa of the MT. However once, the reservation has been made, if the MT moves to a new M-router rather than modifying the current reserved path to the new MT location, the reserved path could be extended from the old M-router to the new M-router.

The benefit of this approach is that the RSVP flow state information in routers between the CN and the old M-router should not need to be modified. Flow state only needs to be created on the path between the old M-router and new M-router. The other benefit of this approach is that the end-to-end RSVP level reservation can be achieved very quickly when the MT moves to a new M-router since only a small portion of the path needs to be changed. The technique, also known as "Path Extension" [Ach97], may result in sub-optimal paths if the MT executes consecutive inter-subnet handovers with the same connections-flows active.

Although this approach presents some deficiencies regarding non-optimal routing, it may be preferred to other alternatives as the elongation of the existing path is not expected to extend for a large number of hops. As argued in [New97], typical flows which may be managed in a cut-through manner include the ftp-data, telnet, gopher, HTTP, login, audio, etc. Furthermore some of these applications such as telnet may use RSVP to reserve resources to minimise the delay experienced by telnet packets.. As discussed in [Lee97], telnet session duration fits into a log-normal distribution. The average session time is approximately 240 sec = 4 min. Information regarding the statistical analysis of telnet sessions is given in [Pax95]. We assume that a picocell has a radius of 20m and is traversed with walking velocity (2.5 Km/h ~ 0.7m/sec). That yields a mean residence time of 57sec. Dividing the average telnet session time by the mean residence time suggests that the considered flow (telnet session) will be disrupted 4 times due to handovers. From those handovers, it is quite reasonable to claim that only a small percentage may lead to inter-subnet handovers (which involve the re-establishment of RSVP flow state).

Thus the existing path won't extend to a large number of hops. For an increased number of APs per M-router the percentage of handovers which require path extension is even lower [Lin96].

From our initial investigations this approach seems promising. However the technical details of this approach require further study. In particular, how to tunnel the traffic from the old M-router to the new M-router and still allow RSVP messages to be visible so the QoS requirements can be met. Recent work in the IETF has begun to consider how RSVP could inter-operate with tunnels [Ter98]. Further study is required to finalise the details of this approach.

14.4.2 Solving the Resource Reservation Problem

As discussed in an earlier section QoS guarantees can be maintained within the wireless subnet by the MT informing the new M-router of its QoS requirements. However to re-create the reservation at the IP level the RSVP state information needs to be updated. If the new AP unable to meet the QoS requirements of a given flow, rather than dropping the flow the MT can negotiate a lower QoS which can then be communicated at the IP layer in the RSVP case by modifying the contents of the RESV message. If the wireless subnet can meet the QoS requirements but a router on the new path is unable to, the default action of RSVP is to lower the QoS provided to that flow to best effort. It would be better if the MT could negotiate a lower level of QoS without having to reduce the quality to best effort. This issue is outside the scope of this deliverable which focuses on the wireless IP subnet. However, one issue that must be addressed that impacts the wireless subnet is the speed in which the IP level QoS is reserved along the new portion of the path. The remainder of this section describes alternative approaches to this problem.

14.4.2.1 Rutgers Mobile-RSVP Approach

This approach solves the problem of being able to provide the QoS on the new path quickly by maintaining a list of the networks the MT is likely to visit and storing the required QoS information in each of these networks.

Disadvantages of this approach are: (1) the volume of state information that must be stored; and, (2) the constraint that the MT must know which networks it is likely to visit.

14.4.2.2 Trigger new RSVP messages at the CN when the binding update arrives

One problem with re-generating the reservation when the MT moves is that the default average period between transmission of RSVP messages is 30 seconds. If the CN sent new RSVP messages as soon as it receives a binding update this would help the speedy flow of new PATH and RESV messages. This means we still lose the reservation but not for half a reservation period on average plus the round trip time.

14.4.2.3 Reduce the RSVP retransmission period

By default RSVP retransmits control messages every 30 seconds. One alternative would be to reduce this retransmission period so that the time it takes to re-generate reservations is reduced. The disadvantage of this is that it increases the volume of RSVP control messages.

14.4.2.4 Employ the Path Extension Routing Approach

If the path extension approach described above is employed, the end-to-end RSVP reservation will be regenerated quickly. This is because flow state only needs to be created/modified on the path between the old M-router and the new M-router.

14.4.3 Proposed Solution

The only two feasible solutions to the routing problem appear to be:

1. To pass up the outer header source and destination address information to the RSVP module at every router. This means RSVP implementation changes at the router.

2. Change RSVP and MIP at the CN so that path messages contain the coa. Let RSVP translate the home address to the coa. This requires changes to RSVP and a new MIP interface. This requires modification to end-points only, not intermediate routers.

Independent of the routing approach another new MIP interface to RSVP is needed to trigger the transmission of path messages when a binding update is received to ensure the speedy reservation of resources on the new path.

In the second routing solution a new RSVP EXCHANGE message or new detection mechanism is also needed at routers, to identify that a flow with a new coa is the same as the previous flow, it is just the MT address that has changed. (This is not necessary in the first solution since each router maintains a mapping between home address and care of address). Hence we recommend the first approach, because it requires no change to the RSVP specification. In the second case, to avoid the over reservation problem a new message would need to be defined.

Finally the path extension approach also seems to be a partial solution to the routing problem, and also eases the speedy reservation of resources problem. Hence it needs to be considered further.

14.4.4 Inter-Subnet Hand-Over Operation

During inter-subnet handover, QoS guarantees are not preserved as shown in the Message Sequence Chart in Figure 53. The reservation of resources in the new path will not take effect immediately, causing QoS traffic to be temporarily exchanged over the BE channel.

If QoS connections exist from the CN to the MT, the former will issue PATH messages upon reception of the Binding Update message. The Binding Update message is a prerequisite for the transmission of PATH messages by the CN, as its Binding Cache has to be updated in accordance to MT’s change of coa. The MT will respond with RESV messages. If QoS connections exist in the opposite direction, the MT issues, upon reception of a Binding Update Acknowledgement, RSVP PATH messages towards the CN for updating/installing state information in the intermediate routers. The CN will react with RESV messages. Like the intra-subnet hand-over process, the same CAC procedure for the wireless link in the new subnet and on the new router is performed. Fixed CAC is triggered by RSVP. The two re-establishments (MT to CN, CN to MT) can be overlapped and not necessarily executed sequentially as show in Figure 53.

Two important reasons favour the above solution. Firstly, the forwarding mechanism guarantees that all packets arrive at the new location of the MT. The drawback here is a temporary increase in delay. However, this is mostly dependent on the IP-in-IP encapsulation performance of mobile IP routers, and can be optimised in this respect. Secondly, the period of BE transmission of data towards the mobile is very short if the above described interaction between Mobile IP and RSVP is implemented. Binding updates (and the associated acknowledgements) trigger the regeneration of PATH messages if there are reserved flows. This interaction requires only a small interface added to the RSVP demons and it guarantees that flows recover their QoS requirements after at least 1.5 round-trip delays between CN and MT.

Intermediate routers that are present on both the old and new path should update their flow state information to reflect the MTs new coa when they receive the new PATH messages. New routers on the path between the MT and CN must create the flow state and reserve the requested resources. Since the routers maintain a mapping between the MTs home address and current coa, (as described in the previous sections), this means that the old flow state can be maintained and new RSVP sessions do not need to be created end-to-end.

A separate issue is how to remove the reservation on the old portion of the path between the MT and CN which is not common with the new PATH. RSVP provides timers, so eventually this reservation will be removed. RESVTEAR messages should not be employed because the reservation only needs to be removed on a small portion of the path, not at all routers between the MT and CN.

Note that although the reservation can not be made at the new M-router at the IP level until PATH and RESV messages are exchanged, QoS can be provided within the wireless IP system, by the MT informing the new M-router about its current resource requirements at a radio level.

To summarise, in the RAN the following minor modifications can help to improve inter-subnet hand-over significantly:

- ND (Neighbour Discovery) is slightly extended for efficient address allocation. By using a positive feedback based on M-routers that track all MT in a subnetwork, duplicate address detection can be reduced from several seconds to a short local round-trip delay of a few milliseconds. It is also important to note that if the uniqueness of MAC addresses can be ensured, e.g. via link level verification, the DAD procedure is not required at all.
- M-routers can forward packets that arrive after the MT has moved to the new M-router, reducing potential packet loss during inter-subnet hand-over.

- QoS can be reserved quickly within the wireless network even when moving between M-routers. This can be achieved by the previous M-router or the MT telling the new M-router what the radio level QoS should be.

In the rest of the Internet enhancements are also possible. The first enhancement improves performance but is not essential. In contrast the second enhancement is essential to ensure that RSVP can correctly interact with MIP. The only other alternative is to modify the CN to send PATH messages containing the coa rather than the home address, and, to extend the RSVP specification to add a new RSVP EXCHANGE message.

- Mobile IP and RSVP may interact to minimise QoS degradation during hand-over. This can be achieved by modifying the CN so that the arrival of a binding update triggers the generation of new PATH messages, and hence the update of the RSVP session between the CN and the MT at its new location.
- RSVP implementations at routers are modified so that the RSVP daemon receives both the RSVP control messages and the outer IP addresses. This enables these routers to maintain a mapping between the MT home address and coa, and hence maintain the RSVP session, even if the MT moves to a new subnet.

15. WIRELESS IP MULTICAST

The wireless interface causes specific requirements for multicast scheme. As a result, both the network layer and the radio link layer must be modified to meet these requirements. This is shown in the following figure. The radio link layer is in charge of multicast traffic delivery in the local network. It also provides a link layer addressing scheme for the groups. This is done by the MultiCast Agents (MCA) added to the radio sub-system of the AP and the MT. The network layer is responsible for routing multicast traffic to other networks. It also provides a way to trace group members in its local network. These are the functionalities of the MCAs in the control block of the MT and the M-router. These issues are described further in the following sections.

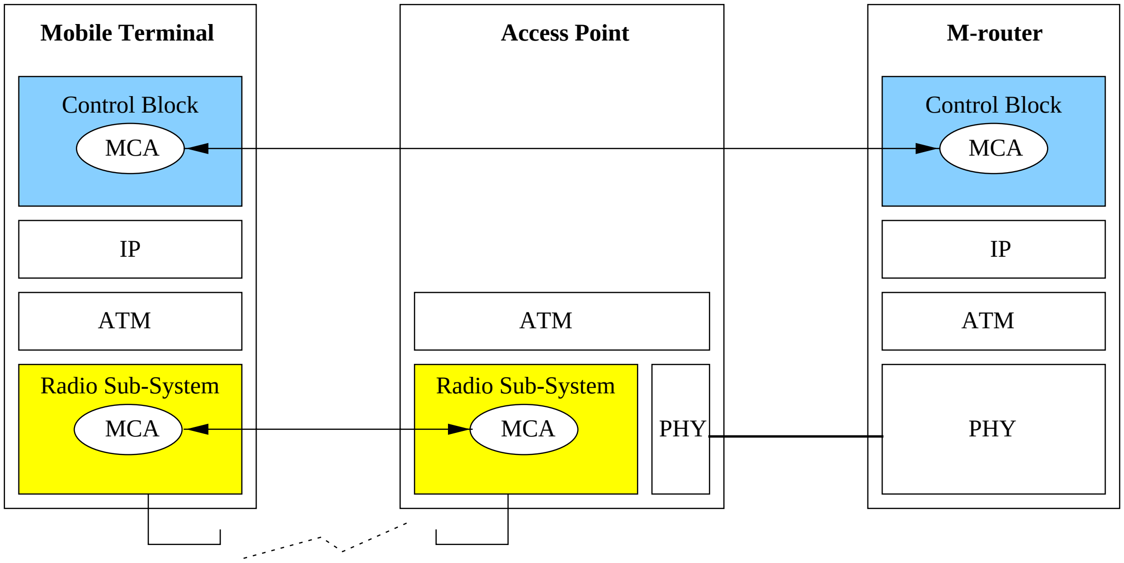


Figure 46: *Multicast architecture*

15.1 Radio Link Layer Requirements

15.1.1 Link Layer Group Addressing

In order to avoid the use of long addresses, each MT is identified by a 2 bytes MAC address in the link layer. This MAC address is given by the AP to the MT during the association phase and is unique in the range of a single AP. Similarly, each AP is identified by a MAC address [3D2].

In the same way, a multicast group is identified by a MAC address in an AP. A MAC address is allocated dynamically to a group whenever there is a MT in an AP, wishing to join a group and that there is no other member of the group present in the AP. This address will be used to identify the group as long as there is at least one member of the group in the AP. Since the address allocation is dynamic, the AP must inform the MTs in its coverage area about the MAC address associated to this group. The members of the group must memorise this MAC address for their further use.

15.1.2 Local Delivery of Multicast Traffic

Group traffic is delivered in a local network by the link layer using the broadcast nature of the radio medium. The concept of a connection identifier has been already introduced in the system for unicast communications. This 1 byte identifier called the MAC Virtual Channel (MVC) is granted to each connection during the connection set-up. At the AP, a mapping is done such as the pair (MT MAC address, MVCI) would correspond to a VPI/VCI in the AP M-router link. The same concept can be applied to the multicast communications. Multicast connections are identified by a MVC in the radio interface. The main question is that how the MVCs are allocated for traffic coming from different sources to the same group inside an AP. Is it possible to use a single MVC for all communications within a group in an AP? In order to answer the above questions, let’s consider the MVC allocation in downlink and uplink directions for the switched data as well as best effort data.

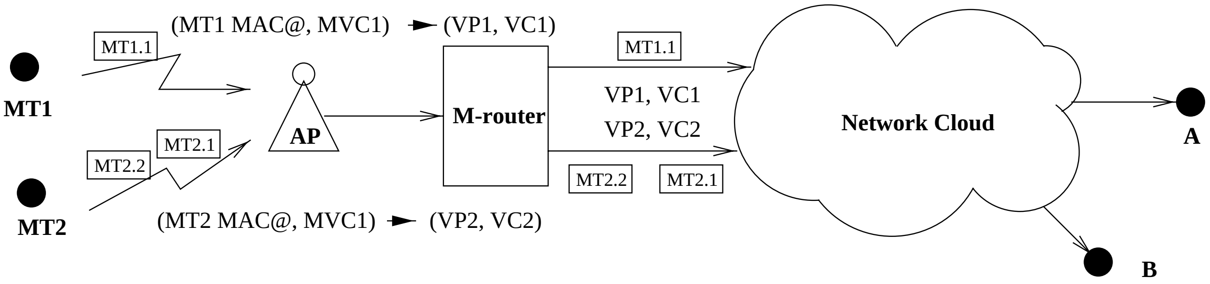


Figure 47: MVC allocation in the uplink

Figure 47 shows the uplink scenario. Group G is composed of the mobile nodes MT1 and MT2 and the fixed nodes A and B. MT1 and MT2 are located in the same AP. Let’s assume also that they begin to send data to the group G at the same time. The use of the same MVC by MT1 and MT2 does not cause any problem in the uplink direction since the uplink data contains the corresponding MT MAC address in the source address field. Therefore the AP can easily distinguish the data coming from different sources.

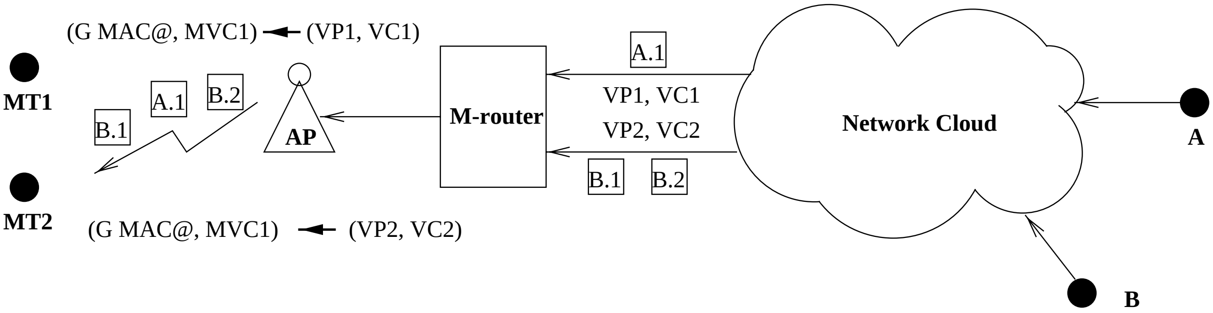


Figure 48: MVC allocation in the downlink

Figure 48 depicts the downlink scenario. Again, the mobile nodes MT1, MT2 and the fixed nodes A and B form a group G. MT1 and MT2 are in the same cell. Let’s assume that A and B have redirected their flows, therefore their data pass directly through ATM link layer via dedicated VCs. The data arrives at the M-router on two different links. Here the use of the same MVC by the AP leads to multiplexing problem in the receivers. The receiver nodes MT1 and MT2 have no way to differentiate the data coming from A and B. This is due to the fact that cells coming from different AAL5 frames can not be interleaved on the same outgoing link.

Now let’s consider the same downlink scenario with best effort data. The cells coming on best effort channels are reassembled in the M-router level. Therefore the AP receives all the cells belonging to the same AAL5 frame sequentially. The use of the same MVC by the AP does not lead to any multiplexing problem in the receivers since the whole AAL5 frame is sent sequentially on the radio interface.

In conclusion, we propose to use one MVC per source for each redirected flow in downlink direction and one MVC for best effort channel (BE MVC). The BE MVC must be created for a group as soon as the AP allocates a MAC address to the group. This connection is used to carry the best effort data as well as the control data to the group members inside an AP. For the redirected flow, however, the AP must allocate a MVC in a dynamic way. It must also inform the group members about the allocated MVC. The BE MVC can be used to communicate the allocated MVC to the group members.

15.2 Network Layer Requirements

15.2.1 Group Membership Management

The Multicast IP standard uses the IGMP protocol [Dee89] in order to keep track of group members in a local network. For this purpose, each multicast IP router needs to know the presence of a group in its local network. A multicast packet will be broadcast in a local network if the specified group has at least one member in that local network. A wireless network requires more considerations for multicast. In order to avoid the waste of bandwidth, multicast packets must be forwarded only to the cells with active members of the specified group. This necessitates the M-router to keep more information than the list of present groups in its local network. In our system the M-router maintains a list of APs per group in order to avoid tracing each MT individually. A similar approach has been proposed in [Ach96]. We associate a location list L_g to a group g as follows:

An AP belongs to L_g if at least one MT belonging to g is located in its cell.

Whenever there is traffic coming for group *g*, the M-router consults the group location list L_g. It then forwards the group traffic only to the APs found in L_g. In this way, a multicast packet is not transmitted to the APs that do not have any member of the group. Since the radio is a shared medium, all the multicast packets can be sent without addressing the recipients explicitly. This mechanism needs only to know the APs to which the multicast packet must be transmitted. Besides, if we track the location of each MT individually, every move by each MT will cause a location update in the data base while our AP list is updated less frequently.

The following changes will cause an update in L_g :

- Changes due to host mobility

  Lets assume that h is a MT locating in the AP A and that h is a member of the group g. The Lg will be modified only if:

1. h leaves the AP A and that there is no other member of g in A. In this case A must be deleted from Lg.
2. h enters another AP B where there is no member of g. B must be added to the list Lg.

- Changes due to group membership

  Lets assume again that h is a MT locating in the AP A and that h is a member of the group g. Lets also assume that h' is a MT locating in the AP A'. The Lg will be modified only if:

1. h leaves the group g and there is no other member of g in A. In this case A must be deleted from Lg.
2. h' joins to the group g and no other member of g exists in A'. A' must be added to the list Lg.

Our group membership protocol is based on an explicit join/leave mechanism. Since the APs are not IP aware in our system, all the join/leave messages must be sent between MT and M-router. A MT sends a join or a leave message to the M-router whenever it wants to join or leave a group. In order to update its database, the M-router needs to detect the following situations in each AP:

- First join, a MT sends a join demand for a group that has no other member in its AP.
- Last leave, a MT, which is the only member of a group in its AP, sends a leave message for the group.

These situations can be detected by associating a *reference number* to each group. The reference number $R_{g,\,A}$ is defined as the number of members of the group *g* in AP *A*. Initially the reference number of all groups in each AP are zero. The reference number of zero means that there is no member of the group in the specified AP. Whenever a join message is received for a group *g* in an AP *A*, the reference number $R_{g,\,A}$ is incremented. The reference number is decremented upon reception of a leave message.

As a conclusion, the M-router maintains the *location list L* as defined below :

G is the list of all existing groups: {g}
A is the list of all APs attached to the M-router.
R_g *is the list of reference numbers of group g in each AP :* $\{R_{g,A}\}$
L_g *is the location list of group g.*
$L = \{ (L_g , R_g) \mid g \in G \}$

Finally, it is important to note that we do not treat the problem of multicast address allocation in our group membership protocol. Applications like Session Directory (SD) can be used for this purpose.

15.2.2 Global Delivery of Multicast Traffic

Global multicast traffic delivery is assured by the multicast routing protocols. The earliest routing mechanism is DVMRP (Distance Vector Multicast Routing Protocol) [Wai88]. This protocol is based on the distance vector algorithm and builds a multicast tree per source. A second proposal, MOSPF (Multicast Open Shortest Path First) [Moy94] is based on the OSPF protocol which uses a link state algorithm. This protocol also builds a multicast tree per source. The last proposal is the CBT (Core Based Tree) [Bal93] which employs a single tree for each group rather than one tree per source. The description of these protocols is beyond the scope of this document.

Our multicast scheme does not have a compatibility problem with these multicast routing protocols. For example, DVMRP is the protocol used in the MBone. It builds a multicast tree per source using the information stored in multicast routers. This information consists of the list of all present groups in the local network of the routers. In our system, this information can be easily found from the location list *L* of the M-router.

15.3 The Overall Mechanism

A MT subscribes itself to a group by sending a join message to the M-router. A join process requires the following steps:

- A MT wanting to join a group g sends a join message to the M-router using the multicast IP address of g.
- The M-router determines if the MT, that issued the join message, is the first member of g in its AP by means of reference number Rg, A. If this is the case, the M-router must ask the corresponding AP to allocate a MAC address for g. The AP must also open a permanent MVC for the best effort channel.
- The AP then communicates the MAC address of g to the MT.
- The MT adds the MAC address of g to its group list. It also opens a permanent MVC for the best effort channel.
- If the MT is the first member of g in its AP, the M-router adds the AP MAC address to the location list of g.
- The M-router increments the reference number Rg, A.
- If there are active multicast connections, the MT must be informed about the MVCs allocated to them.

A MT unsubscribes itself from a group by sending a leave message to the M-router. The following steps must be done when a MT wants to leave a group.:

- A MT wanting to leave a group *g* sends a leave message to the M-router with the multicast IP address of *g*.
- The MT deletes *g* from its group list. It kills the group best effort channel and all the active connections.
- The M-router decrements the group reference number $R_{g, A}$.
- The M-router determines if the MT is the last member of *g* in its AP by means of $R_{g, A}$. If this is the case, it asks the corresponding AP to releases the MAC address allocated to *g* and to kill all its active connections as well as the best effort connection.
- If the MT is the last member of *g* in its AP, the M-router deletes the AP MAC address from the location list of *g*.

The MSCs of the join and leave processes can be found in annex A.

15.4 The Impact of Mobility on Multicast

We consider two cases to study the effect of mobility on multicast. The first case is when the MT makes a handover to an AP within the same domain; intra-domain handover. The second case is when the MT moves to an AP in another network; inter-domain handover. Inter-domain handover follows the mobile IP [Joh98] approach where an MT gets a temporary address known as care of address while away from its home network.

15.4.1 Intra-Domain Handover

In general, we can identify two intra-domain handover schemes: forward and backward. Forward handover scenario occurs when the MT loses its association with its old AP. Therefore, it implies that all of the handover signalling is performed through the new AP. In backward handover scheme, the MT has still its connection with its old AP and therefore, the old AP can be used for the exchange of mobility signalling information.

In both handover schemes, the MT must inform the M-router about the groups to which it belongs. The M-router must identify these groups in order to update their reference numbers. If a group has no other member in the old AP, its MAC address as well as all its active connections must be released. The M-router must also delete the old AP from the group location list. On the other hand, the new AP must allocate new MAC addresses to those groups that have no members in its cell. In this case, the M-router must add the new AP to the location lists of those groups. The complete signalling exchange for both forward and backward handovers can be found in the appendix.

15.4.2 Inter-Domain Handover

When entering a foreign network, the MT must inform the local multicast router about the groups to which it belongs. The multicast router executes the group membership protocol and delivers the multicast traffic to the MT directly. The advantage of this scheme is its optimal routing, efficient bandwidth consumption and its natural support for multicast. The multicast traffic can be delivered taking advantage of the native multicast capacity of the wireless networks. However the MT risks to loose packets during handover. When a MT that belongs to a group *g*, enters a foreign network with no member of *g*, it can not receive the data sent to *g* immediately even though it had already joined *g*. The reason is that the local multicast router needs to graft a path to multicast trees for *g* with respect to all active sources. This introduces a delay d_{graft} which is the time required for a router to attach itself to the multicast tree. To this delay, we must also add the time that it takes for the corresponding AP to allocate a new MAC address for the group in question and to inform the MT of the new MAC address. Therefore, the MT can not receive the multicast traffic before this delay. In the case where the MT enters a network that already has a member of *g*, the latency is very low. The packet losses due to the delay d_{graft} can be avoided by the previous multicast router acting as a home agent and forwarding the multicast traffic to the MT. The previous multicast router must also be informed about the multicast groups of the MT. In case where the MT was the only member of a group, the previous multicast router must prune itself from the corresponding multicast trees.

In the same way, the MT sends data to a multicast group directly on the foreign network using its care of address as the source address. This approach is optimal in bandwidth use and routing. However there are the applications that rely on source addresses to identify streams from different senders such as video conferencing. The recipients can not know that although two multicast datagrams contain different source addresses, they originated from the same mobile sender which has moved across different networks. This problem can be easily resolved by the use of home address destination option of IPv6. Therefore all multicast datagrams sent by a MT in a foreign network use the MT's care of address as the source IP address and contain the MT's home address in the home address destination option. However, when the MT is moving fast between different networks, the cost of building a multicast tree may not be acceptable.

15.5 Reliable Multicast

To assure reliable multicast communication the following issues must be considered:

- Reliable transmission of data
- Reliable end-to-end communication

The reliable transmission of data is assured by the data link layer. This functionality is necessary for wireless links due to their high bit error rate and packet loss. End-to-end related tasks are those that are processed at the end systems only. This is usually the case for ordering, synchronisation, and packet drops due to network congestion. These end-to-end tasks can be performed either in the application layer or in the transport layer.

15.5.1 Reliable Transmission of Multicast Data

Within our radio link layer, the Wireless Data Link Control (WDLC) entity is responsible for error control over the medium. For loss sensitive services, an ARQ protocol is used to guarantee error free transmission, performing retransmission of erroneously delivered data. Retransmission is applied only for the non-real-time traffic. For that purpose WDLC uses a Go-Back-N protocol with a sliding window size of M=16 [3D2].

The WDLC entity can be extended to support the multicast communications. The same ARQ mechanism is used to retransmit the original data for the non-real-time multicast traffic. For the time-constrained data, no reliability can be guaranteed because of the imposed delay of the error recovery techniques.

An ARQ mechanism relies on its receivers feedback in order to decide to retransmit the data. However in a multicast communication, the retransmission depends on the state of all the receivers. If each MT (receiver) tries to send its own feedback to the AP, a feedback implosion may occur as it is depicted in the following figure. In order to avoid the feedback implosion in the AP, we have chosen a redundant feedback type in which one feedback message is sufficient per group in each AP.

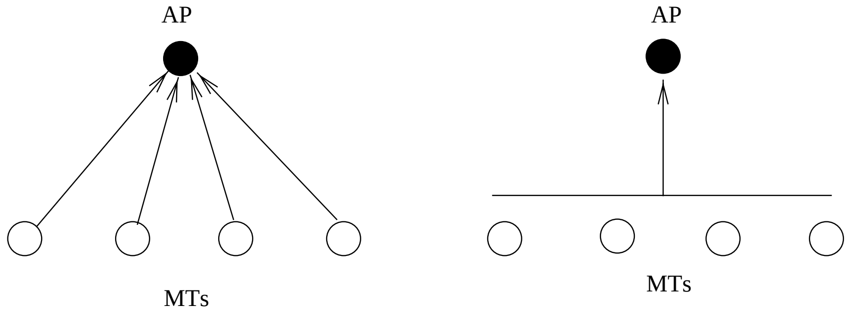


Figure 49: Different feed back types for group communication

This scheme needs a mechanism of feedback suppression as explained below:

- · At MT
  - On loss detection, choose a random number and listen for other’s feedback (NAK).
  - On reception of feedback from another group member, cancel timer and suppress feedback.
  - On timer expiration, send a feedback to the AP using the group MAC address as source address.
- · At AP
  - On reception of a feedback, multicast it to the group address in order for all group members located in the same cell to listen to it.

In case of error detection, the AP retransmits the lost packet to all group members. The advantage of this scheme is when there are several MTs in an AP that have lost the same packet. They will receive the lost packet with only one retransmission. However, the other group members receive the lost packet even if they have received it correctly.

As a conclusion, our multicast error recovery scheme is as follows:

- An ARQ mechanism is used for error control,
- The original packet will be retransmitted in case of error,
- One feedback per group is sent to each AP from the members local to that AP,

- The lost packet is multicast to all group members of an AP.

15.5.2 Reliable End-to-End Multicast Communication

Many reliable transport protocols have been developed for multicast communication such as XTP (Xpress transport Protocol) [XTP95], SRM (Scaleable Reliable Multicast) [Flo95], and RMTP (Reliable Multicast Transport Protocol) [Pau97]. However some researchers propose providing the reliability function at the application level. Due to the lack of a standard reliable transport protocol, most IP multicast applications have their own error control mechanism. Some examples of the multicast applications are listed below. The main design principles, common to all of these applications is that end-to-end control is provided at the application level and not via a general purpose multicast transport protocol [Dio97].

1. Audio and Video Conferencing, some examples of this kind of application are vic [McC95], vat [Vat96], rat [Rat96] and freephone [Bol96]. These applications do not require any reliability or transport level ordering. Audio packets are reordered in the application play-out buffer. Application layer congestion control is also generally implemented.
2. Shared Workspace, wb (whiteboard) [Wb96] is an example of such an application. Wb provides reliable multicast. Application level recovery is performed if an out of order packet is received.

Given that end-to-end reliability functions are handled by the majority of applications themselves, we do not believe that a reliable multicast transport stack is required in the wireless IP system.

15.6 QoS Control

The problem of QoS has not yet been completely solved for unicast communication. Moreover, unicast QoS solutions can not always be adapted to group communication where each participant may have its own constraints which are not necessarily acceptable to the other participants. The discussion of QoS control for multicast communication is independent from the QoS solution that is used in the backbone network. Here the main issue is how to control the QoS for multicast communication given that a shared medium is used for data transmission.

We propose to define the QoS of a group as the minimum of each member's QoS. Each MT sends its QoS request to the M-router via whatever mechanism is available to the network. The M-router then chooses the minimum of each member's QoS as the QoS for the whole group. However, this means that all group members will be penalised if there exists only one receiver which does not have enough capacity for a higher level of QoS.

16. CONCLUSIONS

This deliverable describes how the WAND broadband radio technology can be modified to support best effort and real-time unicast and multicast Internet applications. This document has described a system based upon IPv6. An integral part of the system is the mobility enhanced router. This is an extension of a standard IPv6 router. By default all traffic is carried in a best-effort fashion. However the M-router can identify flows that should receive a higher level of service based on the volume of packets seen for a given packet stream, or on the arrival of RSVP messages. These flows are marked with a dedicated flow identifier that can be mapped to a radio flow identifier. Three service classes are supported at the radio layer, and each flow is mapped to the appropriate priority class based on the QoS requirements and traffic characteristics of the flow. This approach inter-operates with both Integrated and Differentiated Services approaches for real-time traffic support in the Internet.

The system also supports multicast applications. To achieve this the M-router keeps track of which access points have MTs which are members of each multicast group. Multicast traffic for that group is then only forwarded to these access points. To avoid broadcasting traffic to all MTs associated with a given access point, multicast MVCs are defined. This ensures the traffic only reaches those MTs which are members of the multicast group.

This document has also described mobility mechanisms for both within a wireless IP subnet and when the MT moves between two M-routers. In the intra IP subnet case the M-router manages the switching of control and data connections from the old access point to the new access point. Inter-subnet mobility is supported by an extension of the Mobile IPv6 protocol. In this case the MT must obtain a new care of address. This causes difficulties if RSVP is employed at the IP level. The deliverable has described solutions to this problem that can be achieved via minor modifications to current RSVP implementations.

To summarise this deliverable has provided a specification for a broadband wireless IP network. Support for real-time and best-effort unicast and multicast applications is provided even if the terminal moves between different wireless IP subnets.

17. ANNEX OF MSCS

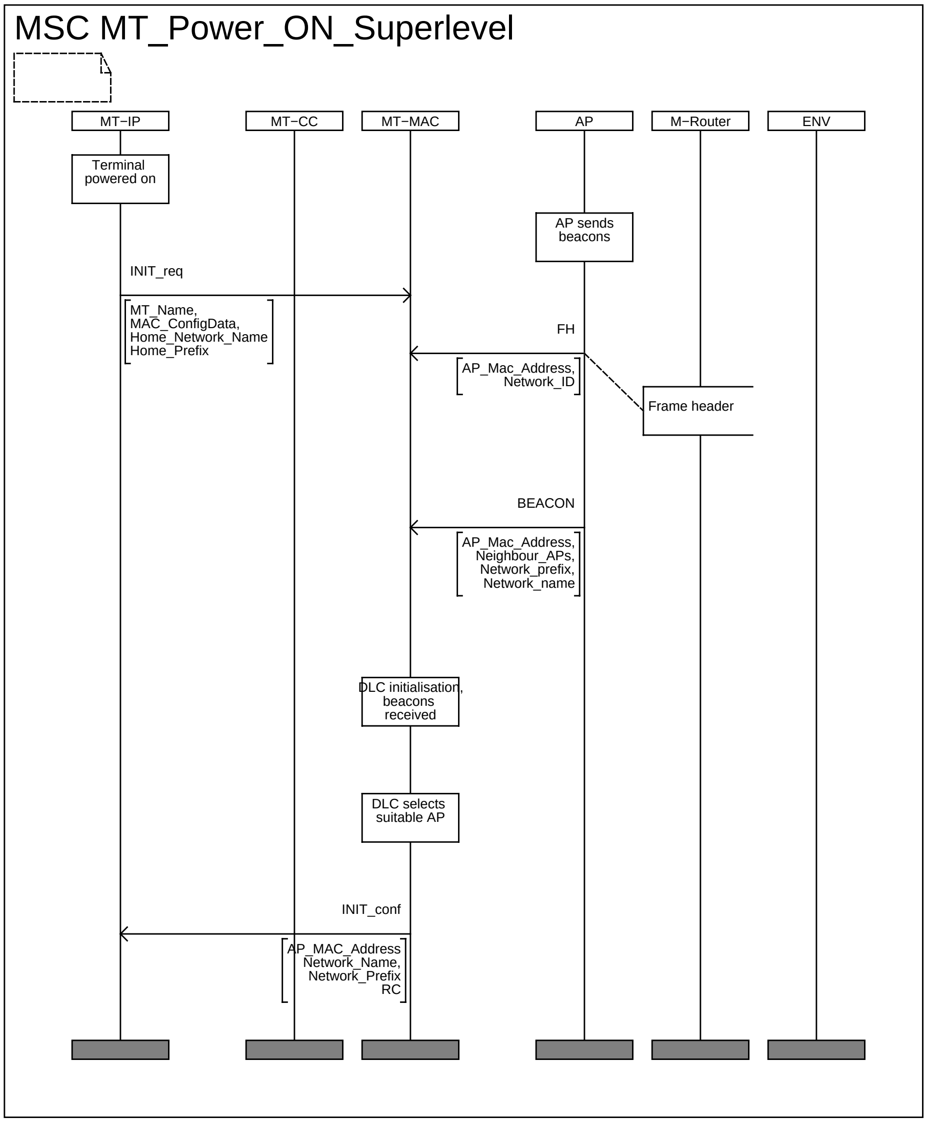

MSC MT_Power_ON_Superlevel
MT−IP
MT−CC
MT−MAC
AP
M−Router
ENV
Terminal powered on
AP sends beacons
INIT_req
MT_Name, MAC_ConfigData, Home_Network_Name Home_Prefix
FH
AP_Mac_Address, Network_ID
Frame header
BEACON
AP_Mac_Address, Neighbour_APs, Network_prefix, Network_name
DLC initialisation, beacons received
DLC selects suitable AP
INIT_conf
AP_MAC_Address Network_Name, Network_Prefix RC

Figure 50: Message Sequence Chart for MT Power On

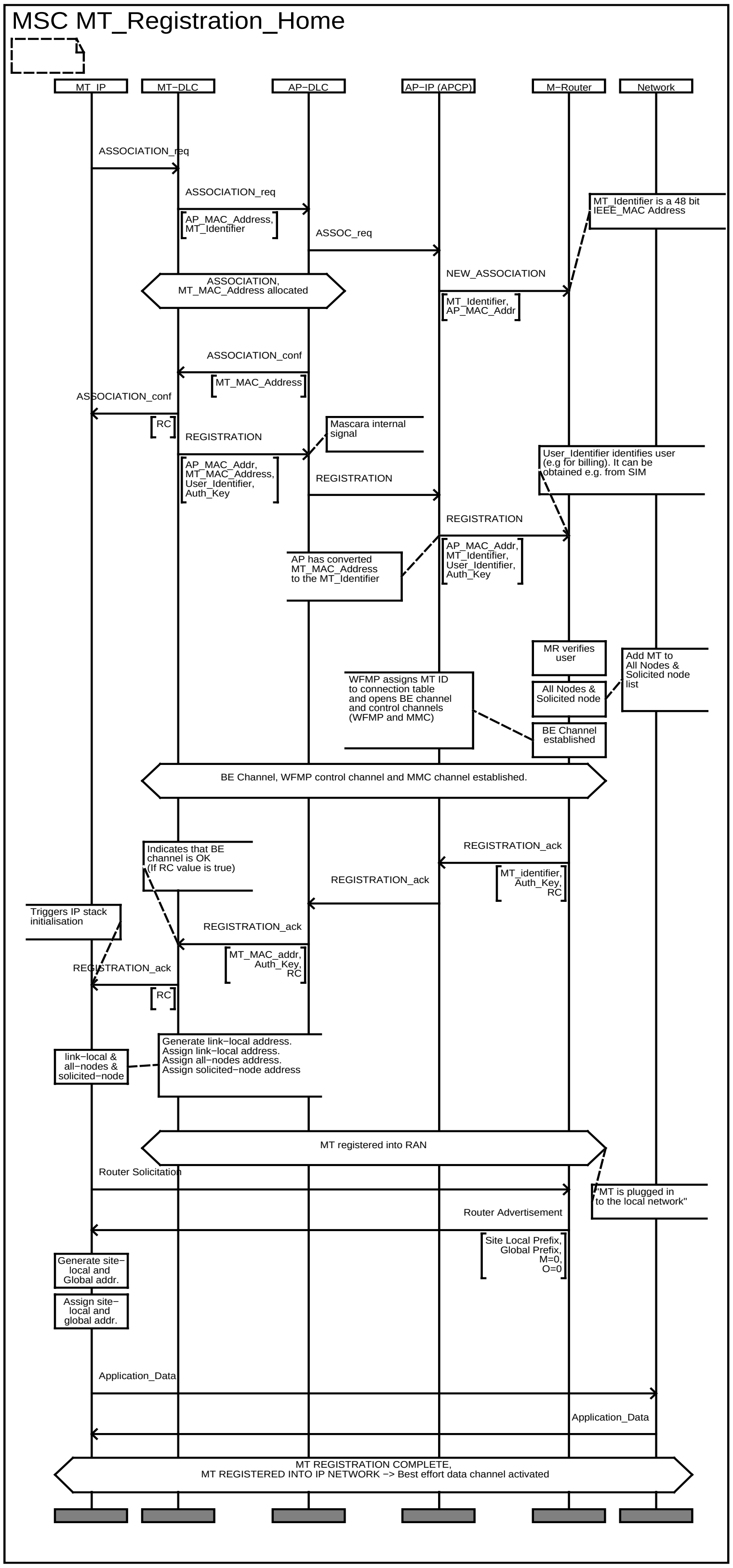


Figure 51: Message Sequence Chart MT Registration at Home

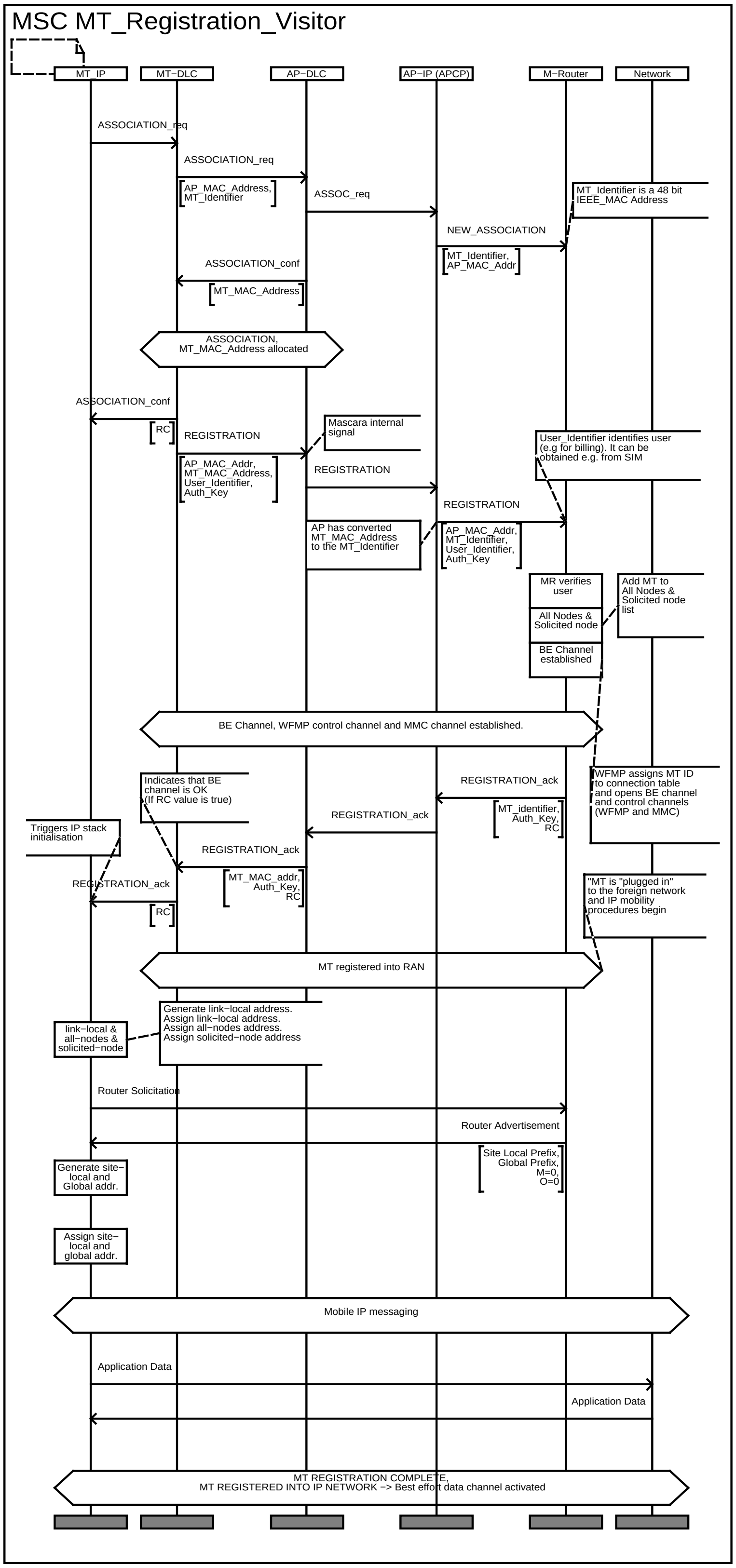

MSC MT_Registration_Visitor
MT_IP
MT−DLC
AP−DLC
AP−IP (APCP)
M−Router
Network
ASSOCIATION_req
ASSOCIATION_req
AP_MAC_Address, MT_Identifier
ASSOC_req
MT_Identifier is a 48 bit IEEE_MAC Address
NEW_ASSOCIATION
MT_Identifier, AP_MAC_Addr
ASSOCIATION_conf
MT_MAC_Address
ASSOCIATION, MT_MAC_Address allocated
ASSOCIATION_conf
RC
REGISTRATION
Mascara internal signal
AP_MAC_Addr, MT_MAC_Address, User_Identifier, Auth_Key
REGISTRATION
User_Identifier identifies user (e.g for billing). It can be obtained e.g. from SIM
REGISTRATION
AP has converted MT_MAC_Address to the MT_Identifier
AP_MAC_Addr, MT_Identifier, User_Identifier, Auth_Key
MR verifies user
All Nodes & Solicited node
BE Channel established
Add MT to All Nodes & Solicited node list
BE Channel, WFMP control channel and MMC channel established.
Indicates that BE channel is OK (If RC value is true)
REGISTRATION_ack
WFMP assigns MT ID to connection table and opens BE channel and control channels (WFMP and MMC)
REGISTRATION_ack
MT_identifier, Auth_Key, RC
Triggers IP stack initialisation
REGISTRATION_ack
MT_MAC_addr, Auth_Key, RC
REGISTRATION_ack
RC
"MT is "plugged in" to the foreign network and IP mobility procedures begin
MT registered into RAN
Generate link−local address. Assign link−local address. Assign all−nodes address. Assign solicited−node address
link−local & all−nodes & solicited−node
Router Solicitation
Router Advertisement
Site Local Prefix, Global Prefix, M=0, O=0
Generate site−local and Global addr.
Assign site−local and global addr.
Mobile IP messaging
Application Data
Application Data
MT REGISTRATION COMPLETE, MT REGISTERED INTO IP NETWORK −> Best effort data channel activated

Figure 52: Message Sequence Chart for MT Registration in a Visited Network

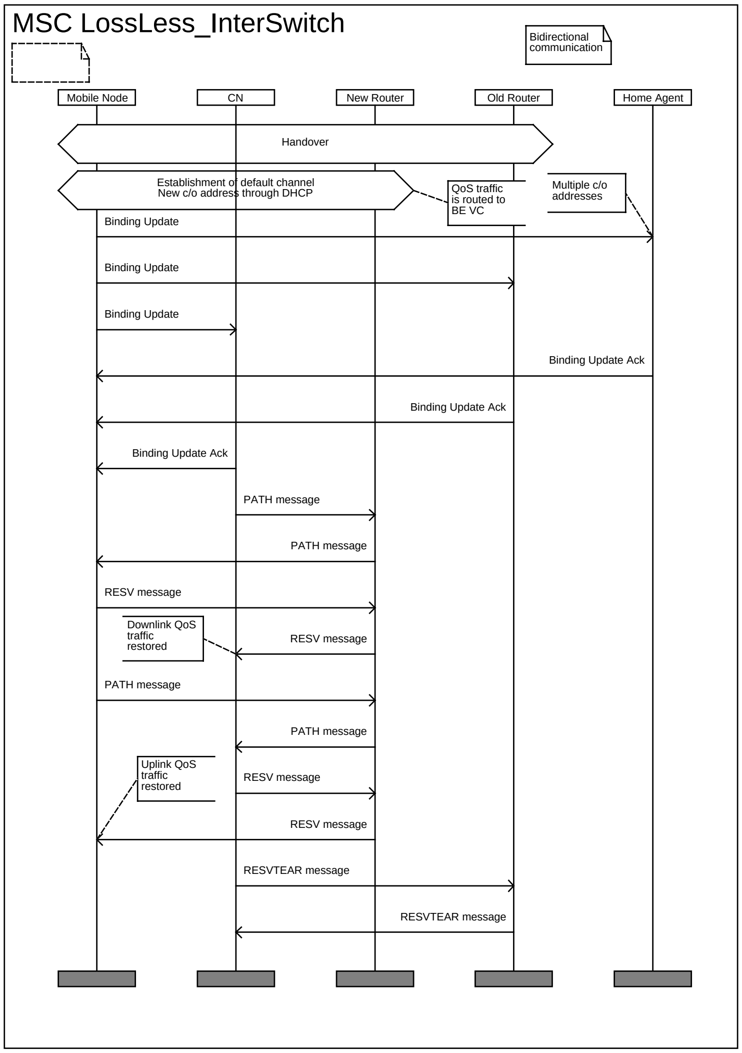


Figure 53: Message Sequence Chart for Inter-IP subnetwork handover

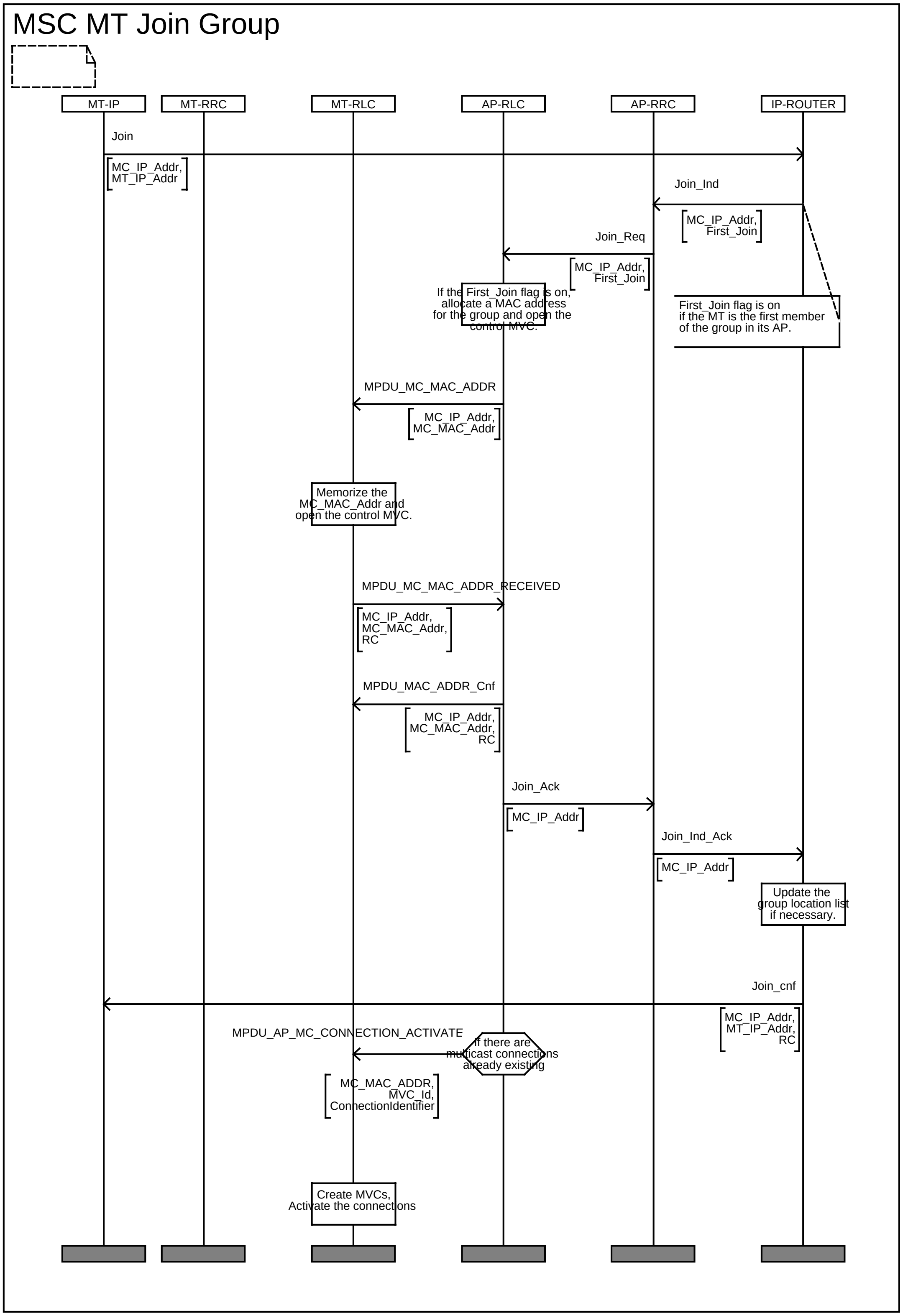

MSC MT Join Group
MT-IP
MT-RRC
MT-RLC
AP-RLC
AP-RRC
IP-ROUTER
Join
MC_IP_Addr, MT_IP_Addr
Join_Ind
MC_IP_Addr, First_Join
Join_Req
MC_IP_Addr, First_Join
If the First_Join flag is on, allocate a MAC address for the group and open the control MVC.
First_Join flag is on if the MT is the first member of the group in its AP.
MPDU_MC_MAC_ADDR
MC_IP_Addr, MC_MAC_Addr
Memorize the MC_MAC_Addr and open the control MVC.
MPDU_MC_MAC_ADDR_RECEIVED
MC_IP_Addr, MC_MAC_Addr, RC
MPDU_MAC_ADDR_Cnf
MC_IP_Addr, MC_MAC_Addr, RC
Join_Ack
MC_IP_Addr
Join_Ind_Ack
MC_IP_Addr
Update the group location list if necessary.
Join_cnf
MC_IP_Addr, MT_IP_Addr, RC
MPDU_AP_MC_CONNECTION_ACTIVATE
If there are multicast connections already existing
MC_MAC_ADDR, MVC_Id, ConnectionIdentifier
Create MVCs, Activate the connections

Figure 54: Message Sequence Chart for MT Joining a Multicast Group

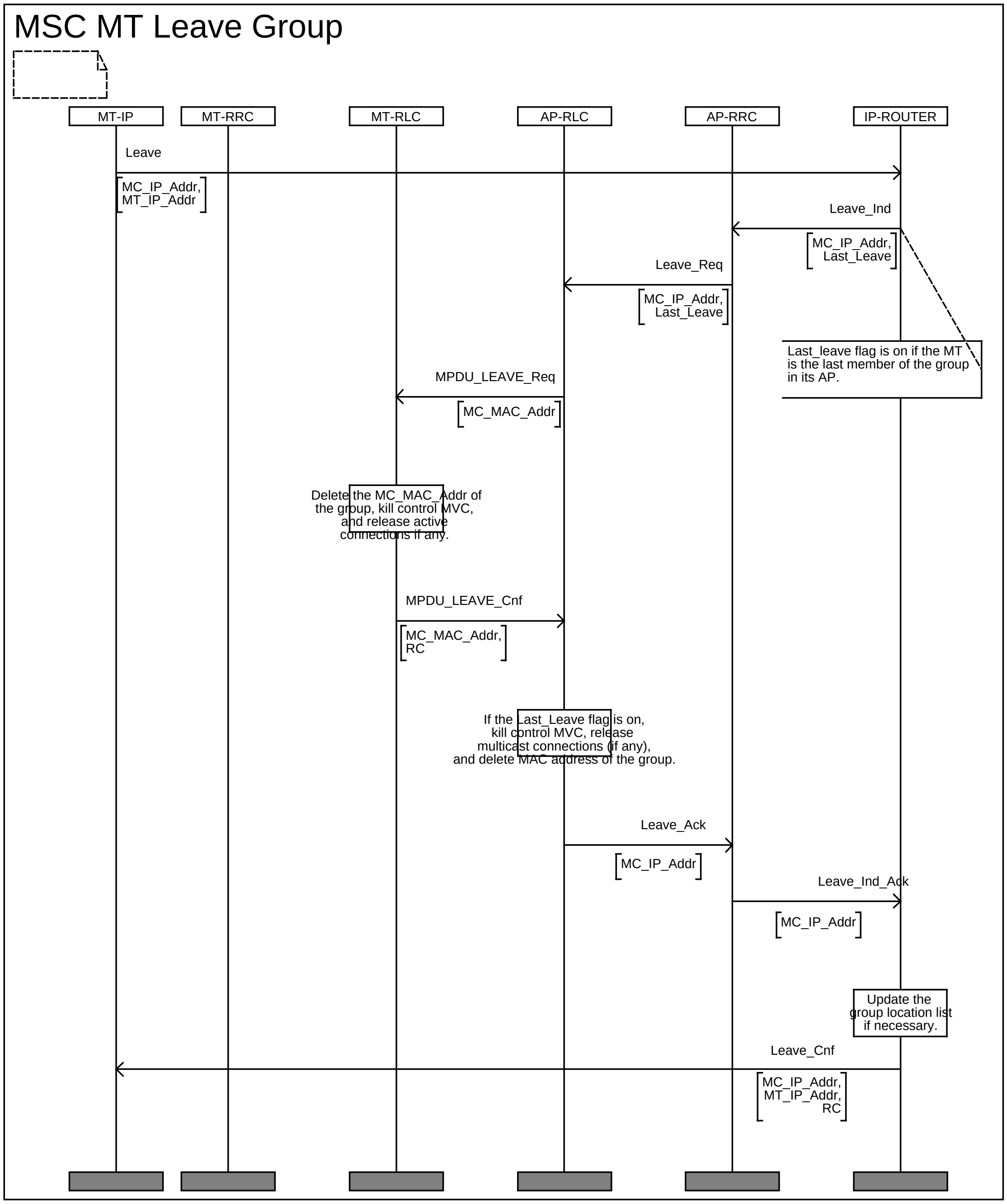
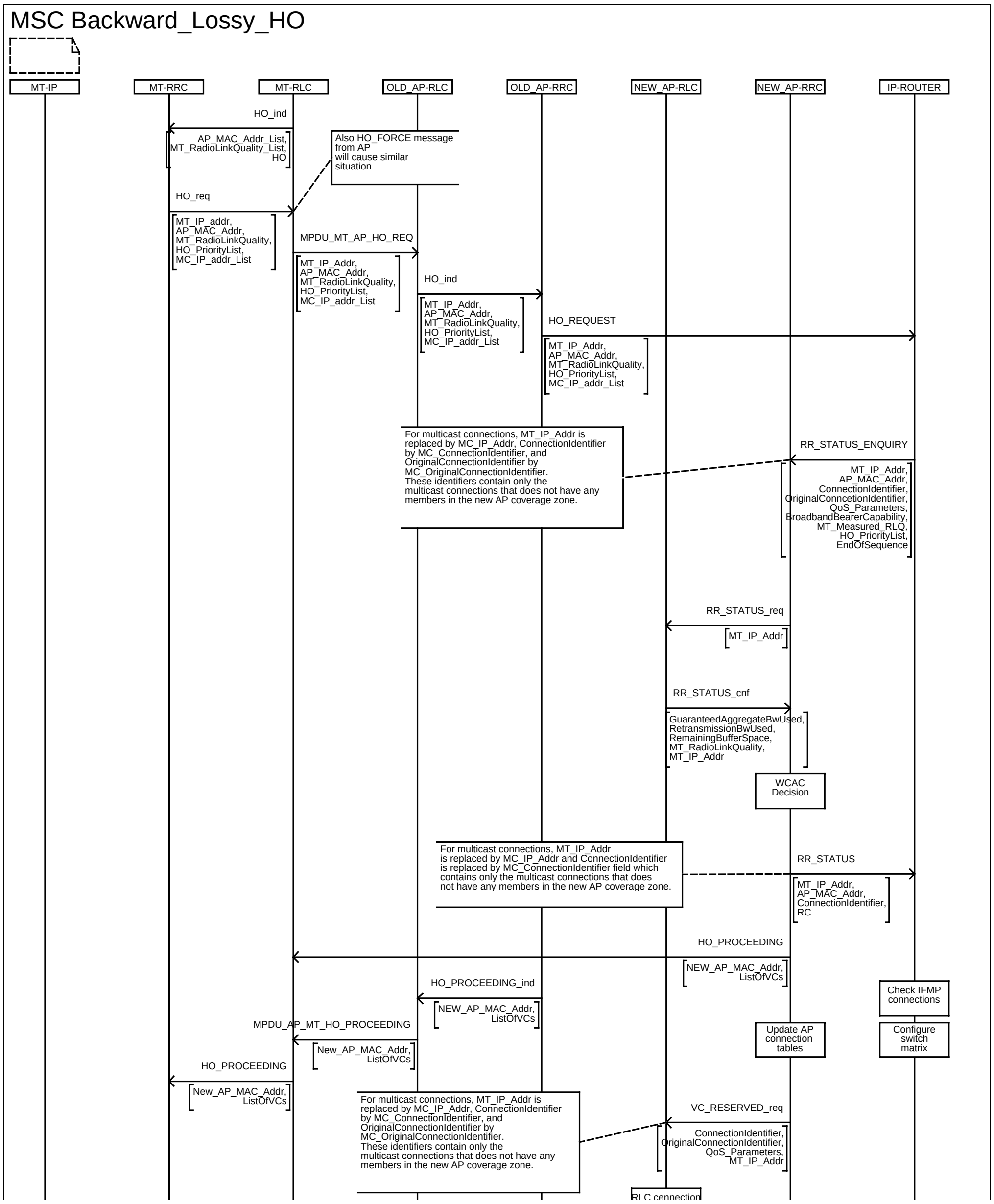

MSC MT Leave Group
MT-IP
MT-RRC
MT-RLC
AP-RLC
AP-RRC
IP-ROUTER
Leave
MC_IP_Addr, MT_IP_Addr
Leave_Ind
MC_IP_Addr, Last_Leave
Leave_Req
MC_IP_Addr, Last_Leave
Last_leave flag is on if the MT is the last member of the group in its AP.
MPDU_LEAVE_Req
MC_MAC_Addr
Delete the MC_MAC_Addr of the group, kill control MVC, and release active connections if any.
MPDU_LEAVE_Cnf
MC_MAC_Addr, RC
If the Last_Leave flag is on, kill control MVC, release multicast connections (if any), and delete MAC address of the group.
Leave_Ack
MC_IP_Addr
Leave_Ind_Ack
MC_IP_Addr
Update the group location list if necessary.
Leave_Cnf
MC_IP_Addr, MT_IP_Addr, RC

Figure 55: Message Sequence Chart for a MT Leaving a Multicast Group

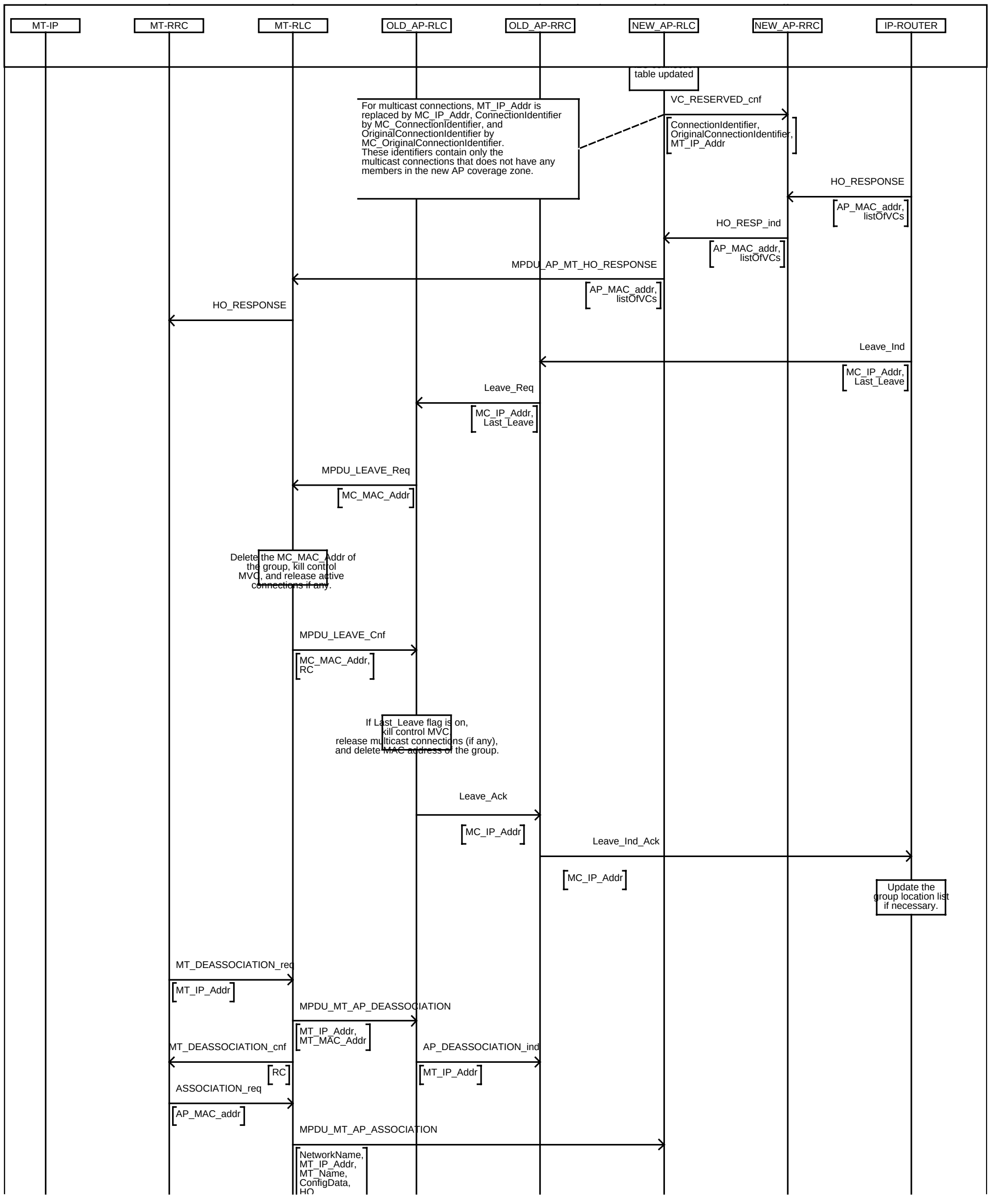

MSC Backward_Lossy_HO
MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
HO_ind
AP_MAC_Addr_List, MT_RadioLinkQuality_List, HO
HO_req
MT_IP_addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_addr_List
MPDU_MT_AP_HO_REQ
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_addr_List
HO_ind
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_addr_List
HO_REQUEST
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_addr_List
RR_STATUS_ENQUIRY
MT_IP_Addr, AP_MAC_Addr, ConnectionIdentifier, OriginalConncetionIdentifier, QoS_Parameters, BroadbandBearerCapability, MT_Measured_RLQ, HO_PriorityList, EndOfSequence
RR_STATUS_req
MT_IP_Addr
RR_STATUS_cnf
GuaranteedAggregateBwUsed, RetransmissionBwUsed, RemainingBufferSpace, MT_RadioLinkQuality, MT_IP_Addr
WCAC Decision
RR_STATUS
MT_IP_Addr, AP_MAC_Addr, ConnectionIdentifier, RC
HO_PROCEEDING
NEW_AP_MAC_Addr, ListOfVCs
Check IFMP connections
HO_PROCEEDING_ind
NEW_AP_MAC_Addr, ListOfVCs
Update AP connection tables
Configure switch matrix
MPDU_AP_MT_HO_PROCEEDING
New_AP_MAC_Addr, ListOfVCs
HO_PROCEEDING
New_AP_MAC_Addr, ListOfVCs
VC_RESERVED_req
ConnectionIdentifier, OriginalConnectionIdentifier, QoS_Parameters, MT_IP_Addr
RLC connection

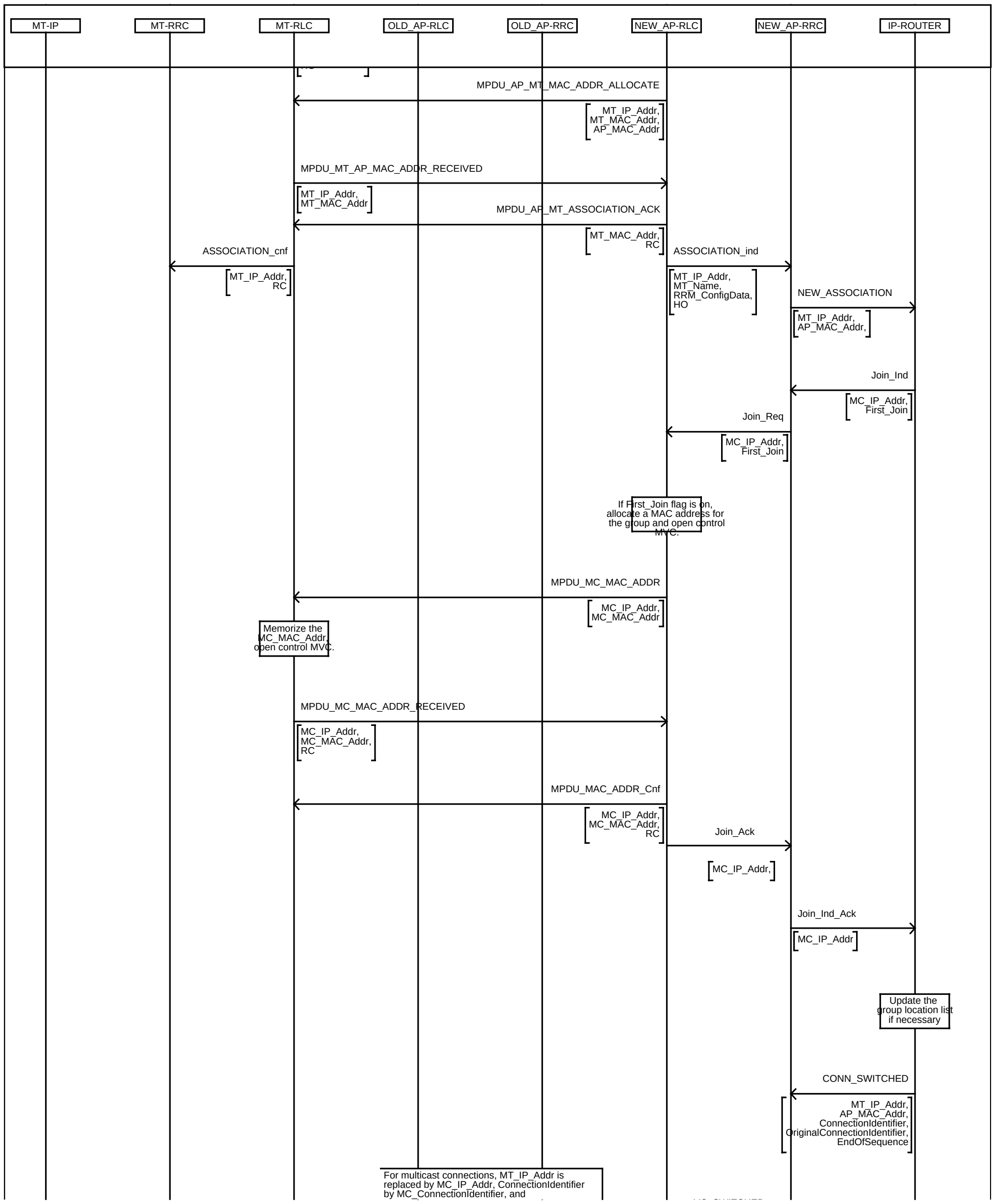
Also HO_FORCE message from AP will cause similar situation

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr and ConnectionIdentifier is replaced by MC_ConnectionIdentifier field which contains only the multicast connections that does not have any members in the new AP coverage zone.

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.


MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
table updated
VC_RESERVED_cnf
ConnectionIdentifier, OriginalConnectionIdentifier, MT_IP_Addr
For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.
HO_RESPONSE
AP_MAC_addr, listOfVCs
HO_RESP_ind
AP_MAC_addr, listOfVCs
MPDU_AP_MT_HO_RESPONSE
AP_MAC_addr, listOfVCs
HO_RESPONSE
Leave_Ind
MC_IP_Addr, Last_Leave
Leave_Req
MC_IP_Addr, Last_Leave
MPDU_LEAVE_Req
MC_MAC_Addr
Delete the MC_MAC_Addr of the group, kill control MVC, and release active connections if any.
MPDU_LEAVE_Cnf
MC_MAC_Addr, RC
If Last_Leave flag is on, kill control MVC, release multicast connections (if any), and delete MAC address of the group.
Leave_Ack
MC_IP_Addr
Leave_Ind_Ack
MC_IP_Addr
Update the group location list if necessary.
MT_DEASSOCIATION_req
MT_IP_Addr
MPDU_MT_AP_DEASSOCIATION
MT_IP_Addr, MT_MAC_Addr
MT_DEASSOCIATION_cnf
RC
AP_DEASSOCIATION_ind
MT_IP_Addr
ASSOCIATION_req
AP_MAC_addr
MPDU_MT_AP_ASSOCIATION
NetworkName, MT_IP_Addr, MT_Name, ConfigData, HO


MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
MPDU_AP_MT_MAC_ADDR_ALLOCATE
MT_IP_Addr,
MT_MAC_Addr,
AP_MAC_Addr
MPDU_MT_AP_MAC_ADDR_RECEIVED
MT_IP_Addr,
MT_MAC_Addr
MPDU_AP_MT_ASSOCIATION_ACK
MT_MAC_Addr,
RC
ASSOCIATION_cnf
ASSOCIATION_ind
MT_IP_Addr,
RC
MT_IP_Addr,
MT_Name,
RRM_ConfigData,
HO
NEW_ASSOCIATION
MT_IP_Addr,
AP_MAC_Addr,
Join_Ind
MC_IP_Addr,
First_Join
Join_Req
MC_IP_Addr,
First_Join
If First_Join flag is on,
allocate a MAC address for
the group and open control
MVC.
MPDU_MC_MAC_ADDR
MC_IP_Addr,
MC_MAC_Addr
Memorize the
MC_MAC_Addr,
open control MVC.
MPDU_MC_MAC_ADDR_RECEIVED
MC_IP_Addr,
MC_MAC_Addr,
RC
MPDU_MAC_ADDR_Cnf
MC_IP_Addr,
MC_MAC_Addr,
RC
Join_Ack
MC_IP_Addr,
Join_Ind_Ack
MC_IP_Addr
Update the
group location list
if necessary
CONN_SWITCHED
MT_IP_Addr,
AP_MAC_Addr,
ConnectionIdentifier,
OriginalConnectionIdentifier,
EndOfSequence
For multicast connections, MT_IP_Addr is
replaced by MC_IP_Addr, ConnectionIdentifier
by MC_ConnectionIdentifier, and

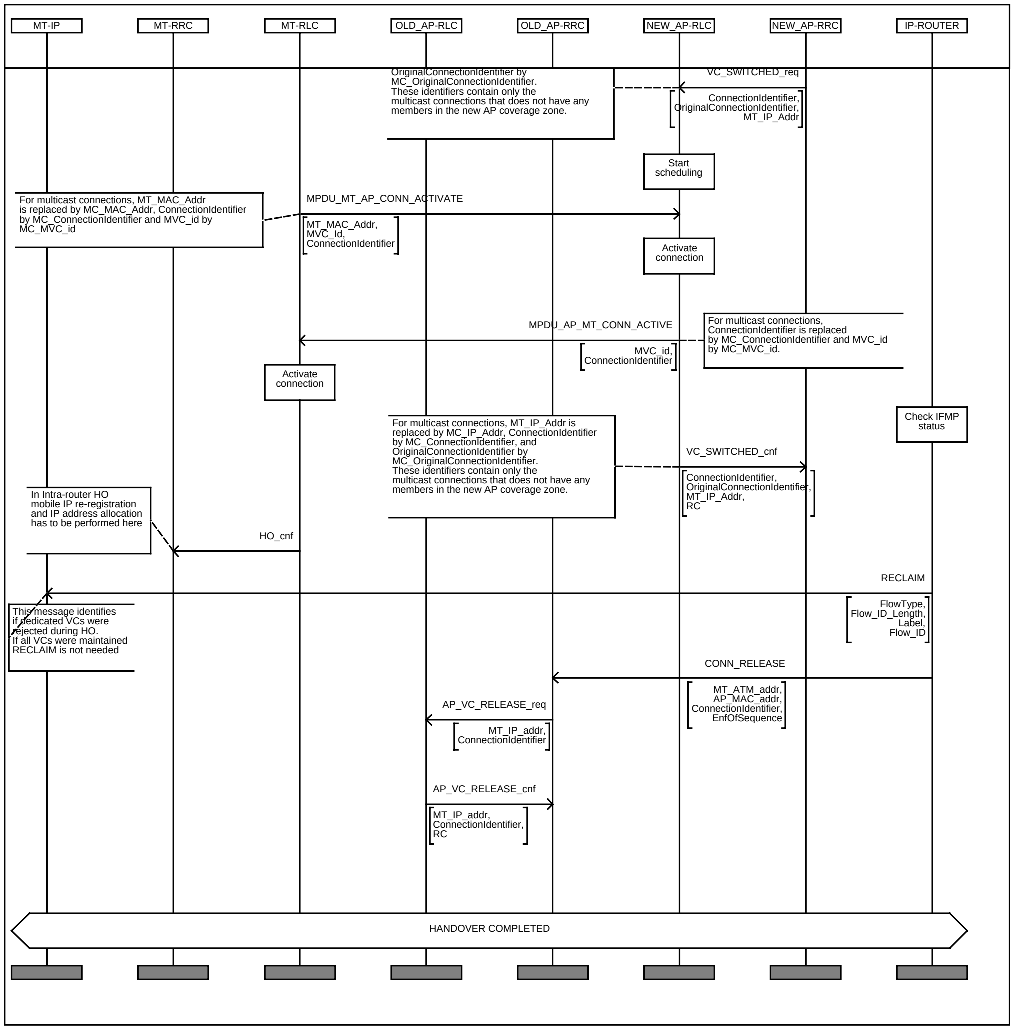
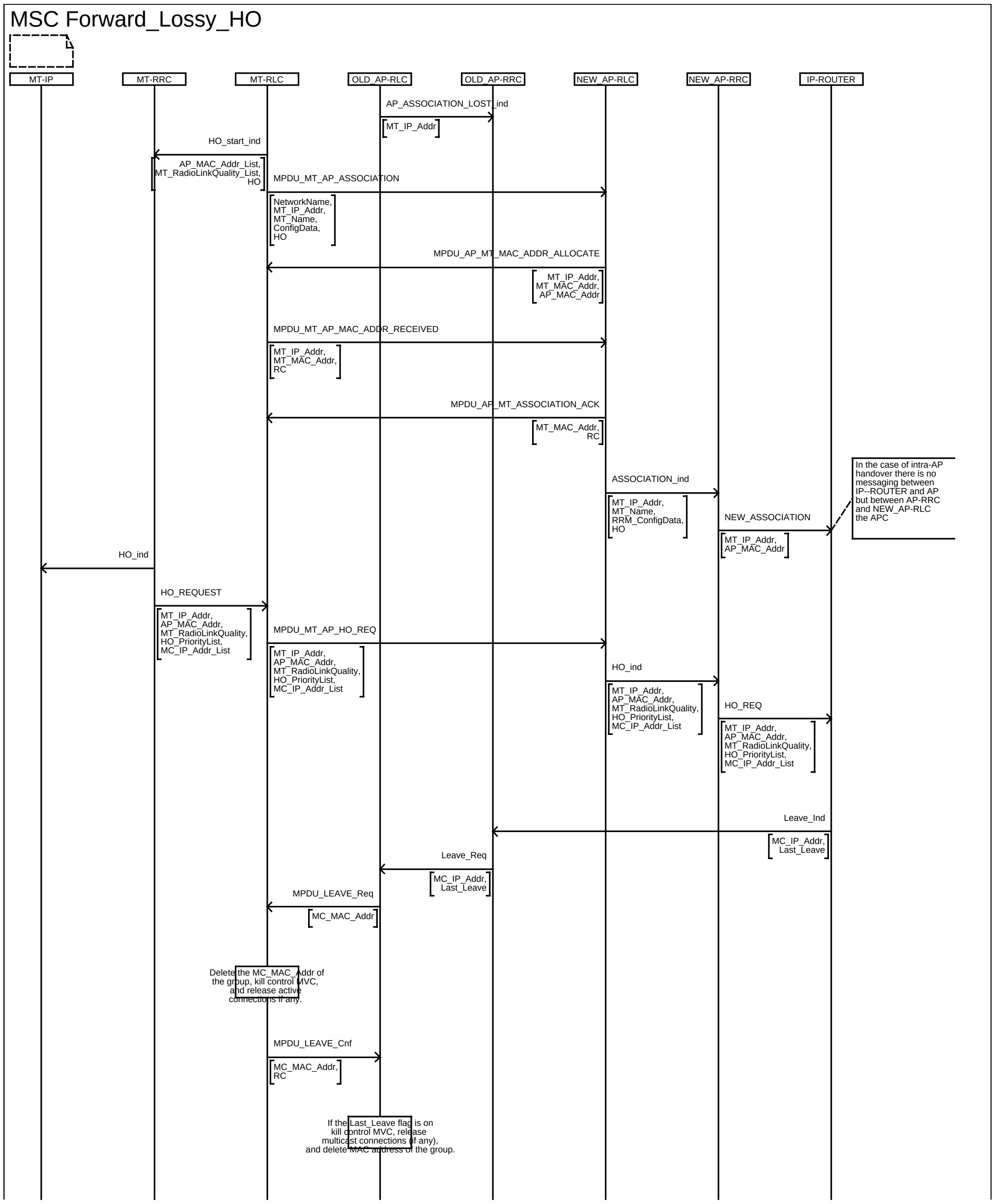
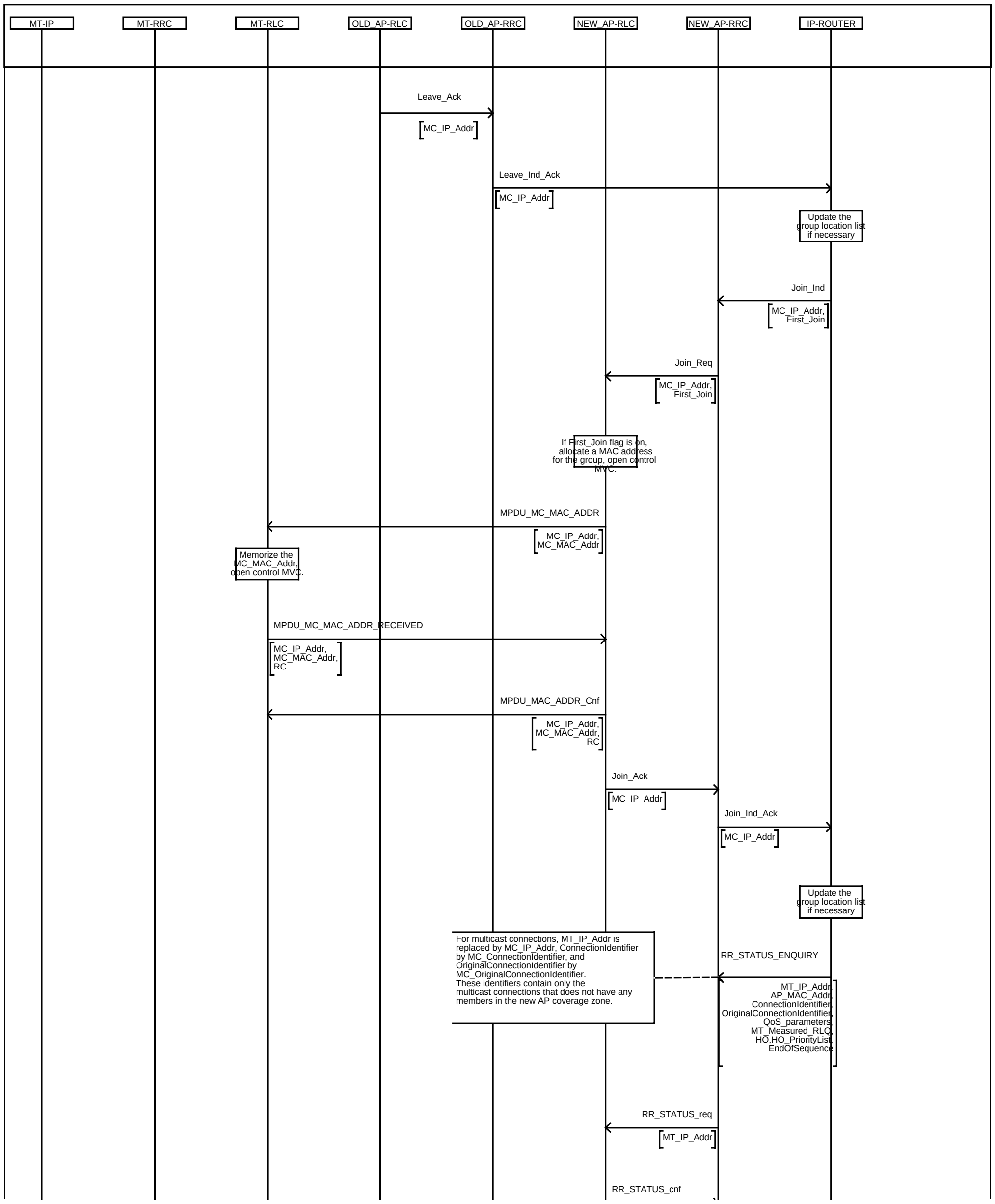
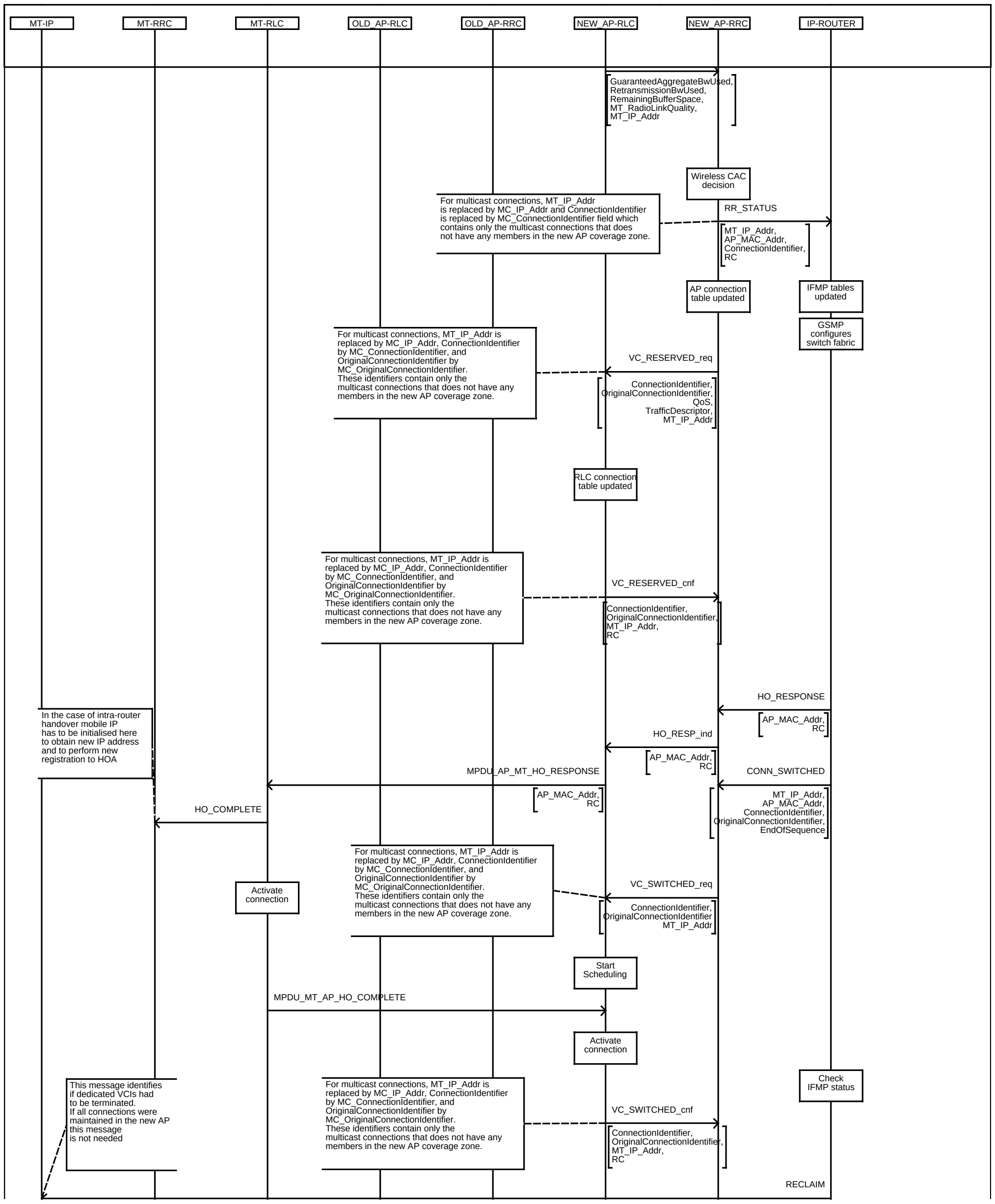

MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
OriginalConnectionIdentifier by
MC_OriginalConnectionIdentifier.
These identifiers contain only the
multicast connections that does not have any
members in the new AP coverage zone.
VC_SWITCHED_req
ConnectionIdentifier,
OriginalConnectionIdentifier,
MT_IP_Addr
Start
scheduling
For multicast connections, MT_MAC_Addr
is replaced by MC_MAC_Addr, ConnectionIdentifier
by MC_ConnectionIdentifier and MVC_id by
MC_MVC_id
MPDU_MT_AP_CONN_ACTIVATE
MT_MAC_Addr,
MVC_Id,
ConnectionIdentifier
Activate
connection
MPDU_AP_MT_CONN_ACTIVE
MVC_id,
ConnectionIdentifier
For multicast connections,
ConnectionIdentifier is replaced
by MC_ConnectionIdentifier and MVC_id
by MC_MVC_id.
Activate
connection
Check IFMP
status
For multicast connections, MT_IP_Addr is
replaced by MC_IP_Addr, ConnectionIdentifier
by MC_ConnectionIdentifier, and
OriginalConnectionIdentifier by
MC_OriginalConnectionIdentifier.
These identifiers contain only the
multicast connections that does not have any
members in the new AP coverage zone.
VC_SWITCHED_cnf
ConnectionIdentifier,
OriginalConnectionIdentifier,
MT_IP_Addr,
RC
In Intra-router HO
mobile IP re-registration
and IP address allocation
has to be performed here
HO_cnf
RECLAIM
FlowType,
Flow_ID_Length,
Label,
Flow_ID
This message identifies
if dedicated VCs were
rejected during HO.
If all VCs were maintained
RECLAIM is not needed
CONN_RELEASE
MT_ATM_addr,
AP_MAC_addr,
ConnectionIdentifier,
EnfOfSequence
AP_VC_RELEASE_req
MT_IP_addr,
ConnectionIdentifier
AP_VC_RELEASE_cnf
MT_IP_addr,
ConnectionIdentifier,
RC
HANDOVER COMPLETED

Figure 56: Message Sequence Chart for Multicast Backward Lossy Handover


MSC Forward_Lossy_HO
MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
AP_ASSOCIATION_LOST_ind
MT_IP_Addr
HO_start_ind
AP_MAC_Addr_List, MT_RadioLinkQuality_List, HO
MPDU_MT_AP_ASSOCIATION
NetworkName, MT_IP_Addr, MT_Name, ConfigData, HO
MPDU_AP_MT_MAC_ADDR_ALLOCATE
MT_IP_Addr, MT_MAC_Addr, AP_MAC_Addr
MPDU_MT_AP_MAC_ADDR_RECEIVED
MT_IP_Addr, MT_MAC_Addr, RC
MPDU_AP_MT_ASSOCIATION_ACK
MT_MAC_Addr, RC
ASSOCIATION_ind
MT_IP_Addr, MT_Name, RRM_ConfigData, HO
NEW_ASSOCIATION
MT_IP_Addr, AP_MAC_Addr
In the case of intra-AP handover there is no messaging between IP--ROUTER and AP but between AP-RRC and NEW_AP-RLC the APC
HO_ind
HO_REQUEST
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_Addr_List
MPDU_MT_AP_HO_REQ
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_Addr_List
HO_ind
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_Addr_List
HO_REQ
MT_IP_Addr, AP_MAC_Addr, MT_RadioLinkQuality, HO_PriorityList, MC_IP_Addr_List
Leave_Ind
MC_IP_Addr, Last_Leave
Leave_Req
MC_IP_Addr, Last_Leave
MPDU_LEAVE_Req
MC_MAC_Addr
Delete the MC_MAC_Addr of the group, kill control MVC, and release active connections if any.
MPDU_LEAVE_Cnf
MC_MAC_Addr, RC
If the Last_Leave flag is on kill control MVC, release multicast connections (if any), and delete MAC address of the group.


MT-IP
MT-RRC
MT-RLC
OLD_AP-RLC
OLD_AP-RRC
NEW_AP-RLC
NEW_AP-RRC
IP-ROUTER
Leave_Ack
MC_IP_Addr
Leave_Ind_Ack
MC_IP_Addr
Update the group location list if necessary
Join_Ind
MC_IP_Addr, First_Join
Join_Req
MC_IP_Addr, First_Join
If First_Join flag is on, allocate a MAC address for the group, open control MVC.
MPDU_MC_MAC_ADDR
MC_IP_Addr, MC_MAC_Addr
Memorize the MC_MAC_Addr, open control MVC.
MPDU_MC_MAC_ADDR_RECEIVED
MC_IP_Addr, MC_MAC_Addr, RC
MPDU_MAC_ADDR_Cnf
MC_IP_Addr, MC_MAC_Addr, RC
Join_Ack
MC_IP_Addr
Join_Ind_Ack
MC_IP_Addr
Update the group location list if necessary
For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.
RR_STATUS_ENQUIRY
MT_IP_Addr, AP_MAC_Addr, ConnectionIdentifier, OriginalConnectionIdentifier, QoS_parameters, MT_Measured_RLQ, HO,HO_PriorityList, EndOfSequence
RR_STATUS_req
MT_IP_Addr
RR_STATUS_cnf



For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr and ConnectionIdentifier is replaced by MC_ConnectionIdentifier field which contains only the multicast connections that does not have any members in the new AP coverage zone.

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.

In the case of intra-router handover mobile IP has to be initialised here to obtain new IP address and to perform new registration to HOA

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.

This message identifies if dedicated VCIs had to be terminated. If all connections were maintained in the new AP this message is not needed

For multicast connections, MT_IP_Addr is replaced by MC_IP_Addr, ConnectionIdentifier by MC_ConnectionIdentifier, and OriginalConnectionIdentifier by MC_OriginalConnectionIdentifier. These identifiers contain only the multicast connections that does not have any members in the new AP coverage zone.

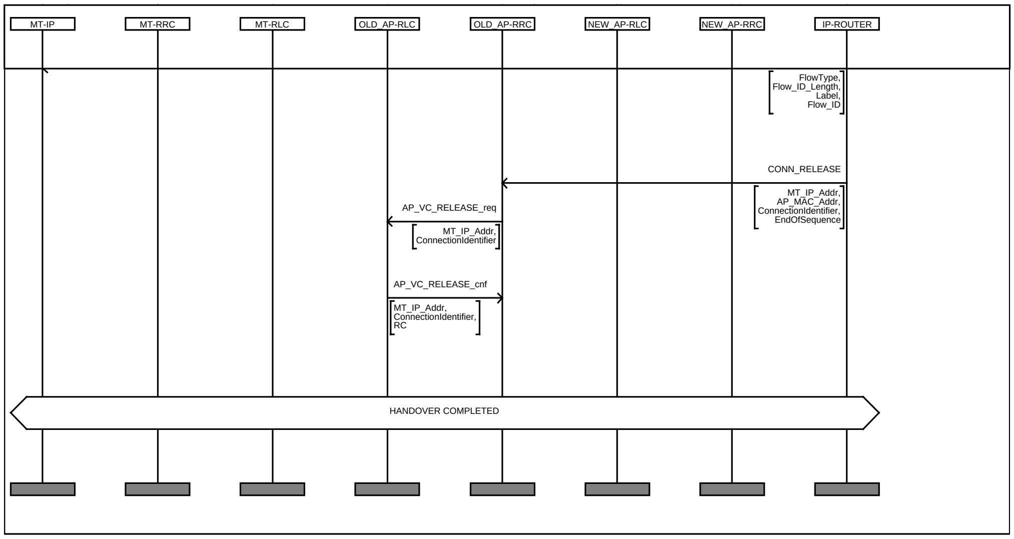


Figure 57: Multicast Forward Lossy Hand-over MSC

18. ANNEX OVERVIEW OF IPV6 CHOICE VS IPV4

This annex describes the basic differences between the Internet Protocol version 4 and version 6 as described by IETF standards documents. Furthermore an outline is given on how these changes from one version to the next affect other protocols, especially those for address management and mobile routing, and how the Internet community is planning to incorporate such extended functionality in the next generation Internet Protocol.
A short introduction on what is new and how the latest IP version developed is given by Bob Hinden in [Hinden95] as follows:

> "IPng was recommended by the IPng Area Directors of the Internet Engineering Task Force at the Toronto IETF meeting on July 25, 1994, and documented in RFC 1752, "The Recommendation for the IP Next Generation Protocol" [1]. The recommendation was approved by the Internet Engineering Steering Group on November 17, 1994 and made a Proposed Standard.
> The formal name of this protocol is IPv6 (where the "6" refers to it being assigned version number 6). The current version of the Internet Protocol is version 4 (referred to as IPv4). This overview is intended to give the reader an overview of the IPng protocol. For more detailed information the reader should consult the documents listed in the reference section.
> IPng is a new version of IP which is designed to be an evolutionary step from IPv4. It is a natural increment to IPv4. It can be installed as a normal software upgrade in Internet devices and is interoperable with the current IPv4. Its deployment strategy was designed to not have any "flag" days. IPng is designed to run well on high performance networks (e.g., ATM) and at the same time is still efficient for low bandwidth networks (e.g., wireless). In addition, it provides a platform for new Internet functionality that will be required in the near future.
> [...]
> IPng includes a transition mechanism which is designed to allow users to adopt and deploy IPng in a highly diffuse fashion and to provide direct interoperability between IPv4 and IPng hosts. The transition to a new version of the Internet Protocol must be incremental, with few or no critical interdependencies, if it is to succeed. The IPng transition allows the users to upgrade their hosts to IPng, and the network operators to deploy IPng in routers, with very little co-ordination between the two."

To give an idea in what time frame IP version 6 will be completed, standardised and finally deployed, several aspects accelerating or delaying the introduction of the next generation IP protocol are discussed. As always, such figures should be taken with a grain of salt. The final question, how the WP1 extension should proceed is addressed in the final section.

18.1 Basic Differences

This section gives a short fact-by-fact introduction and short summary of what has changed between the two versions of IP.

18.1.1 Address Format

In IPv6 the address was extended from 32 bits to 128 bits. IPv4 [RFC791] segments its address space into network classes [RFC796]. Classes A, B and C are allocated for large, medium and small sized organisations, class D for multicast groups and E as reserved test network. In addition to the limited address space of (theoretically) 4 billion addresses classification introduces fragmentation of the address space. In IPv6 [RFC1883] supports more levels of address hierarchies including compatibility addresses for IPv4. An example for an address hierarchy for IPv6 unicast addresses is shown in figure 1 [from RFC1884]:

```
|              n bits            | 80-n bits |     48 bits        |
+--------------------------------+-----------+--------------------+
|        subscriber prefix       | subnet ID |   interface ID     |
+--------------------------------+-----------+--------------------+
<------------------------------128bit----------------------------->

|         s bits        | n bits  |   m bits     | 128-s-n-m bits  |
+----------------------+---------+--------------+-----------------+
|   subscriber prefix  | area ID |  subnet ID   |  interface ID   |
+----------------------+---------+--------------+-----------------+
<------------------------------128bit----------------------------->
```

Figure 1: Example IPv6 Unicast Address Hierarchies

In fact, any number of hierarchy levels may be used. Most importantly, IPv6 addresses can be auto-configured in a simple way by using ICMP (Internet Control Message Protocol), the network address prefix and local MAC addresses [RFC 1971, Stateless Address Autoconfiguration]. Addresses may also be configured with DHCP (dynamic host configuration protocol) which works for both IPv4 and IPv6.

18.1.2 IP Header

Besides the change of address format IPv6 features a new streamlined header which should speed up processing of packets (see figure 2). A new field called the flow label has been introduced in IPv6 to separate individual traffic flows from each other (the label itself is a pseudo random number and is only unique when combined with source and destination addresses and ports). This 20 bit label is intended to be used with reservation schemes such as RSVP (resource reservation protocol) [RFC 1809]. For priority based traffic classification there is an additional 8 bit field called Traffic Class. For both fields, proposal for handling and allocation exist but are not fixed at this time. The payload length is still a 16 bit quantity in IPv6, however there is a new header extension which allows for a 32 bit packet length. The header checksum field is dropped and must be applied in higher layer protocols such as UDP (user datagram protocol) over the whole packet if needed. Fragmentation has been dropped from the header too and is now, in IPv6, processed as an extension header. Furthermore, there are extension headers for hop-by-hop inspection of packets and routing extension headers to define routes for packets explicitly. Finally, security and authentication support in extension headers is also part of the version 6 specification [RFC 1825 through RFC1829].

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Vers=4 |  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
<-----------------------------32bit----------------------------->

 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Vers=6 | Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Header  |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                         Source Address                        +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                      Destination Address                      +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
<-----------------------------32bit----------------------------->
```

Figure 2: IPv4 Header (RFC791) & IPv6 Header (RFC1883 draft-ietf-ipngwg-ipv6-spec-v2-01)

18.2 Migration and Coexistence

Migration of IPv4 to the newer protocol version is managed by reserving the network prefix 0::/96 for the old address space. This format is used to route IPv6 packets over IPv4 routing infrastructure.

```
|                80 bits               | 16 |      32 bits        |
+--------------------------------------+--------------------------+
|0000..............................0000|0000|    IPv4 address     |
+--------------------------------------+----+---------------------+

             IPv4-Compatible IPv6 Address Format

|                80 bits               | 16 |      32 bits        |
+--------------------------------------+--------------------------+
|0000..............................0000|FFFF|    IPv4 address     |
+--------------------------------------+----+---------------------+

             IPv4-mapped IPv6 address
```

Figure 3: Compatible and mapped IPv4 addresses

For IPv4-only hosts the prefix ::FFFF/96 is reserved to be used in IPv6 networks. Migration from version 4 to version 6 will be a transitional process. Hosts running IPv4 can co-exist in an IPv6 network by using the addresses shown in figure 3. Devices using IPv6 in a mixed environment need to use IPv4 tunnelling (as it is done today on the 6-bone) to reach each other and also the compatible addresses to reach IPv4 devices.

18.3 Considerations for Wireless Environments

18.3.1 Addressing

Addressing in IP is for both versions discussed global but with a different address size. What is more important is the handling of address allocation. In current IP networks address allocation is performed mostly manually. DHCP (stateful address configuration) offers a little relief and allows for centralised administration (basically a MAC-address to IP-address database) or even the accommodation of unknown hosts on a subnet. Due to small subnetworks this although very limited. In IPv6 there is a standardised way (stateless address auto-configuration) of obtaining addresses which is built into ICMPv6. Basically the subnet address prefix from router advertisements and the link-address (e.g. a 48 bit MAC address) are combined to yield a valid IPv6 address. This method has the advantage of being available in all protocol stacks (does not need a new protocol and a specialised server) and being reasonable for large numbers of mobile hosts.

18.3.2 Header Size

In wireless systems a low-overhead packet structure is desirable. IPv4 headers are already quite big (20 bytes). In IPv6 this is even worse (40 bytes) but the problem could be solved by applying a simple header compression algorithm (this would reduce the size to at least 12 bytes) or even better, the state information from the wireless link layer could be used to associate packets (which reduces the header to 8 bytes). The advantage of the former method is its transparency, the latter is more efficient but makes some assumptions about another layer in the protocol stack.

Several Internet Drafts discuss this topic (although targeted at low speed serial lines). One proposal [draft-ietf-avt-crtp-03] includes also UDP/RTP headers (additional 20 bytes!) which would be a perfect fit for the WAND applications (WAND audio/video uses RTP on top of AAL5 or UDP). The authors claim a compression of all three headers to 2-4 bytes for many cases.

18.3.3 Mobile IP

Between the Mobile IP for the version 4 of the protocol and the Mobile IPv6 description significant improvements have been made. While the version 4 of Mobile IP tried to accomplish mobility requirements without changing the underlying protocol the new version had some influence of the basic protocol design. Furthermore, the options introduced for mobility support are clearly defined whether they are mandatory for IPv6 implementations or not. For example, general IPv6 stack implementations must only be capable of processing Home Address options to find the right way back to mobiles. Binding updates at correspondent nodes allow for a direct communication with mobile hosts on foreign subnetworks to improve efficiency. Delivery is achieved by using IPv6 routing extension headers rather than by using tunnelling. Routing headers contain a loose source route for the mobile hosts care-of-address. This advanced operation mode must not be implemented by all hosts. For routers which may become home agents, requirements are more stringent. They must be capable of maintaining the bindings for the mobile address, acknowledging binding updates and intercepting and encapsulating packets. Mobile hosts must process or generate binding updates, acknowledges or requests and must be capable of IPv6 decapsulation.

The former use of a foreign agent in IPv4 (router of the foreign subnet) could be avoided in IPv6. The mobile host on the foreign subnetwork performs the binding update (communication with the home agent (router on the home subnet) or with the correspondent node) on its own.

Maximum update frequency for Mobile IP is still the same (1/s) in Mobile IPv6, the protocol is clearly marked as macro mobility management protocol. The draft specification hints at a link layer implementation of micro mobility management for wireless systems with frequent hand-offs.

18.3.4 Security

IPv4 didn't consider security options so far. For this reason, recognising its importance for wireless systems, Mobile IPv4 added authentication for home agents and mobile hosts. For version 6 of Mobile IP, authentication of binding updates and binding ACKs is a must; also a replay protection mechanism is required. These mechanisms are an integral part of IPv6 (authentication and ESP option).

18.4 Market Development

Although IP version 6 is mature and tested from a technical point of view it is very difficult to predict its impact on the market. Given that much work went into the development of a safe and painless transition [RFC 1933] we can assume that it will happen 'as needed'. Users will switch over either if they run out of addresses (which could be quite soon since the static network classes lead to fragmentation of the address space; B and C networks are expected to be sold out first) or if they need some of the new features (security, address allocation). Assuming the current growth of the Internet will continue in a similar fashion, the transition will still take a few years. But if mobile devices (PDAs, mobile phones, pagers) and low-end systems (appliances, set-tops, game-consoles, etc.) will contribute to this growth, the transition could be made much quicker. Developments on the backbone regarding the introduction of IPv6 should be transparent to users of the old version of the protocol thanks to compatible addressing modes.

Current IPv6 implementations include most leading host and router vendors (Apple, Bull, Dassault, Digital, Epilogue, FTP Software, IBM, INRIA, Linux, Novell, Mentat, NRL, NTHU, Pacific Softworks, Process Software, SICS, SCO, Siemens Nixdorf, Silicon Graphics, Sun, UNH, and WIDE, and router implementations by 3Com, Bay Networks, Cisco Systems, Digital, Hitachi Ltd., Ipsilon Networks, Merit, NTHU, Sumitomo Electric, and Telebit Communications [list from http://playground.sun.com/pub/ipng]). Testing started in 1996 on the 6-bone which connects IPv6 hosts and routers though Internet-tunnels in about 30 countries (see also http://www-6bone.lbl.gov/6bone/).

18.5 Conclusions

Table 1 summarises the most important differences for the two IP versions regarding mobility and mobility management. For IPv4 we can generally conclude that mobility was added later. For IPv6 mobility was specified for fixed hosts, routers and mobile hosts. Given that vendors will comply with these specifications it will make the usage of IPv6 very attractive from a commercial point of view. For high-speed wireless access with frequent location updates the micro mobility must still be designed and implemented. Unless immediate compatibility with IPv4 is needed, it would be unwise to invest much power into it. If it is really a requirement, one should try to design and implement the micro mobility in a transparent fashion that would work with either version of IP.

IPv4
compatibility favoured, mobility as add-on
problem of address availability in foreign subnetworks
needs home and foreign agents
packets to mobile host always forwarded/tunnelled
security as an add-on

IPv6
mobility built-in
no addressing problems (big address space and built-in address auto-configuration)
efficient, allows to use direct route from correspondent to mobile
security built-in, but no access policy for foreign networks
may be unsupported (short-term)

Table 1: Comparison of IPv4 and IPv6 features for mobility

18.6 Further IPv6 Reading

S.A.Thomas, IPng and the TCP/IP Protocols: implementing the next generation Internet, John Wiley and Sons, New York, 1996
Robert M. Hinden, IP Next Generation Overview, May 1995
RFC 2133 (IPv6 Sockets)
RFC 1971 (IPv6 Stateless Address Autoconfiguration)
RFC 1970 (Neighbor Discovery for IPv6)

RFC 1933 (Transition Mechanisms for IPv6 Hosts and Routers)
RFC 1885 (ICMPv6)
RFC 1884 (IP Version 6 Addressing Architecture)
RFC 1883 (IPv6 specs)
RFC 1881 (IPv6 Address Allocation Management)
RFC 1825 through RFC1829 (Security, Authentication, Encryption)
RFC 1809 (Using the Flow Label Field in IPv6)
RFC 796 (IPv4 Address Mapping)
RFC 791 (IPv4)
draft-ietf-mobileip-ipv6-04 (November 1997)
draft-ietf-avt-crtp-03 (header compression, July 1997)
draft-ietf-ipngwg-ipv6-spec-v2-01 (base specs IPv6, November 1997)
many other internet drafts upgrading the RFCs above

19. REFERENCES

[Ach96] A. Acharya, and B. R. Badrinath, "A framework for delivering multicast messages in networks with mobile hosts", ACM/Baltzer Journal of Mobile Networks and Applications, vol. 1, No. II, 1996.

[Ach97] A. Acharya, R. Dighe and F. Ansari, *IPSOFACTO: IP Switching Over Fast ATM Cell Transport*, Internet Draft, July 1997.

[Ach98] A. Acharya, J. Li, F. Ansari and D. Raychaudhuri, *Mobility Support for IP over Wireless ATM*, IEEE Communications Magazine, April 1998.

[Ald97] Aldis James, et. Al., “Physical Layer Architecture and Performance in the WAND User Trial System”, ACTS Mobile Communication Summit, Granada Spain, Nov. 1996.

[Ald97] Aldis James, et. Al., “Magic into Reality, building the WAND modem”, ACTS Mobile Communication Summit, Aalborg, Denmark, Oct. 1997.

[ATM95] ATM Forum, ATM User Network Interface (UNI) Specification Version 3.1, Prentice Hall, June 1995.

[Arm96] G. Armitage, ‘Support for Multicast over UNI 3.0/3.1 based ATM Networks’, IETF Standards Track RFC 2022, November 1996.

[Arm97] G. Armitage, "IP multicasting over ATM network", IEEE JSAC, vol. 15, no. 3, April 1997.

[Arm97b] G. Armitage, M. Jork, P. Schulter, G. Harter,’ Transient Neighbors for IPv6 over ATM’ IETF Internet Draft (work in progress), July 1997.

[ATM95] The ATM Forum Technical Committee, ‘LAN Emulation Over ATM’, January 1995.

[ATM96] ‘Ipsilon Steals the Show - INTEROP+NETWORLD 96 - Las Vegas’, The ATM Report, 4(1), pp 6, April 1996.

[ATM97a] The ATM Forum Technical Committee, ‘LAN Emulation Over ATM Version 2 - LUNI Specification’, July 1997.

[ATM97b] The ATM Forum Technical Committee, ‘Multi-Protocol Over ATM Version 1.0’, July 1997.

[Bag95] R. Bagwell, J. McDearman & D Marlow, ‘Achieving IP-Multicast Functionality in the Asynchronous Transfer Mode Environment’, Proceedings of Southeastcon ’95, pp 274-277, March 1995.

[Bal93] A. Ballardie, P. Francis and J. Crowcroft, "Core based trees (CBT)", Proc. ACM SIGCOMM '93, September 1993.

[Bas97] E. Basturk, A. Birman, G. Delp, R. Guerin, R. Haas, S. Kamat, D. Kandlur, P. Pan, D. Pendarakis, R. Rajan, D. Saha and D. Williams, *Design and Implementation of a QoS Capable Switch-Router*, IBM Research Report, RC 20848 (01/31/97).

[Ber97a] L. Berger, ‘RSVP over ATM Implementation Requirements’, IETF Internet Draft (work in progress), July 1997.

[Ber97b] L. Berger, ‘RSVP over ATM Implementation Guidelines’, IETF Internet Draft (work in progress), July 1997.

[Ber98] Y. Bernet, R. Yavatkar, P. Ford, F. Baker, L. Zhang, K. Nichols, & M. Speer, ‘A Framework for Use of RSVP with Diff-Serv Networks’, IETF Internet Draft (Work in Progress), June 1998.

[Bol96] J-C. Bolot, and A. Vega Garcia, "Control mechanisms for packet audio in the Internet", Proceedings of IEEE Infocom '96, 1996.

[Bou98] J. Bound and C. Perkins, “DHCPv6 (Dynamic Host Configuration Protocol Version 6)”, Internet draft, March 1998.

[Bra97] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReServation Protocol (RSVP) - Version 1 Functional Specification", RFC 2205, September 1997.

[Cac96] R. Caceres, V. Padmanabhan, ‘Fast and Scalable Handoffs for Wireless Internetworks’, Proceedings of ACM Mobicom, November 1996.

[Cal97] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, A. Viswanathan, “A Framework for Multiprotocol Label Switching”, IETF Internet Draft (Work in Progress) <draft-ietf-mpls-framework-02.txt>, November 1997.

[Cas97] S. Casner, V. Jacobson, ‘Compressing IP/UDP/RTP Headers for Low-Speed Serial Links’, Internet-Draft (Work in progress), Nov 1997.

[Cla95] K. Claffy, H. Braun, & G. Polyzos, “A Parameterisable Methodology for Internet Traffic Flow Profiling”, IEEE Journal of Selected Areas in Communications, 13(8), October 1995.

[CMU97] CMU, Monarch Project, www.monarch.cs.cmu.edu/mobile_ipv6.html, October 1997.

[Col96] R. Cole, D. Shur & C. Villamizar, ‘IP over ATM: A Framework Document’ IETF Information RFC1932, April 1996.

[Com95] D. Comer, "Internetworking with TCP/IP", vol. 1, Prentice Hall, 1995.

[Cra97] E. Crawley, L. Berger, S. Berson, F. Baker, M. Borden, J. Krawczyk, ‘A Framework for Integrated Services and RSVP over ATM’, IETF Internet Draft (work in progress), July 1997.

[Cro95] J. Crowcroft, Z. Whang, A. Smith & J. Adams, ‘A Rough Comparison of the IETF and ATM Service Models’, IEEE Network, Vol 9(6), pp 12-16, November 1995.

[Dat] Dataman Mobile Computing Lab, “Supporting QoS in Networks with Mobile Hosts, http://athos.rutgers.edu/dataman/qos.html

[Dee89] S. Deering, "Host extensions for IP multicasting", RFC 1112, August 1989.

[Dee95] S.Deering, R.Hinden, “Internet Protocol, Version 6 (IPv6) Specification”, IETF RFC 1883, December 1995.

[Dee96] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu, and L. Wei, "The PIM architecture for wide-area multicast routing", IEEE/ACM Transactions on Networking, vol. 4, no. 2, April 1996.

[Dee97] S. Deering, & R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification”, Internet Draft (Work in Progress), <draft-ietf-ipngwg-spec-v2-01.txt>, November 1997.

[Deg97] M. Degermark, B. Nordgren, & S. Pink, ‘IP Header Compression’, IETF Internet Draft (Work in Progress) <draft-degermark-ipv6-hc-05.txt>, Dec 1997.

[DH76] W. Diffie, M. E. Hellman, “New Direction in Cryptography”, IEEE Transactions on Information Theory, Nov. 1976

[Dif] Differential Service for the Internet, http://diffserv.lcs.mit.edu/

[Dio97] C. Diot, W. Dabbous and J. Crowcroft, "Multipoint communications: a survey of protocols, functions and mechanism", IEEE JSAC, vol. 15, no. 3, April 1997.

[Fen97] W. Fenner, "Internet Group Management Protocol, Version 2", RFC 2236, November 1997.

[Fer97] P.Ferguson, “*Simple Differential Services: IP TOS and Precedence, Delay Indication, and Drop Preference*”, IETF Internet Draft, Work in progress, November 7, 1997.

[Fer98] P. Ferguson, G. Huston, “*Quality of Service, Delivering QoS on the Internet and in Corporate Networks*”, Wiley Computer Publishing, ISBN 0-471-24358-2, January 1998.

[Flo95] S. Floyd, V. Jacobson, S. McCanne, C. G. Liu, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing", Proceedings of ACM SIGCOMM '95, vol. 25, no. 4, August 1995.

[Fly95] S. Flynn, ‘Internet Multicast over Asynchronous Transfer Mode’, MILCOM’95, pp 262-267, Vol 1, 1995.

[Fre97] T. Freeburg, ‘Wireless ATM: Introduction to Mobility and Related Issues’, Presentation at Nomadic ’97, August 1997.

[Fri98] R. Friend, & R. Monsour, ‘IP Payload Compression Using LZS’, Internet Draft (Work in Progress) <draft-ietf-ippcp-lzs-04.txt>, Feb 1998.

[Gar97] M. Garrett & M. Borden, ‘Interoperation of Controlled-Load and Guaranteed-Service with ATM’, IETF Internet Draft (work in progress), July 1997.

[Gha89] M. Ghanbari, Two-layer coding of video signals for VBR networks, IEEE JSAC, 1989

[Got95] Y. Goto, ‘Session Identity Notification Protocol’, IETF Internet Draft (work in progress), July 1995.

[H323] ITU-T Recommendation H.323, “*Visual Telephone Systems and Equipment for Local Area Networks which provide a Non-Guaranteed Quality of Service*”, ITU, 1996.

[Han98] Handley, Schulzrinne & Schooler, ‘SIP: Session Initiation Protocol’, IETF Internet Draft (Work in Progress), June 1998.

[Hei93] J. Heinanen, ‘Multiprotocol Encapsulation over ATM Adaptation Layer 5’, IETF RFC1483, July 1993.

[Hin98] R. Hinden, IPng WG IETF Meeting minutes, Los Angeles, April 1998.

[Hou96] H. Houh, “IP Switching: Server Driven Flow Classification”, Proceedings of the Washington University Workshop on Integration of IP and ATM, November 1996.

[IS] The IETF Integrated Services Working Group charter, http://www.ietf.org/html.charters/intserv-charter.html.

[Jac90] V. Jacobson, ‘Compressing TCP/IP Headers for Low-Speed Serial Links’, Proposed Standard RFC 1144, Feb 1990.

[Jac97] Nichols K., Jacobson V., Zhang L., “*A Two-bit Differentiated Services Architecture for the Internet*”, IETF Internet Draft, Work in progress, November, 1997.

[Joh98] D. Johnson, C. Perkins, 'Mobility Support in IPv6', IETF Internet Draft, Work in Progress, February 1998.

[Kat97] Y. Katsube, K. Nagami, H. Esaki, “Toshiba's Router Architecture Extensions for ATM : Overview”, IETF Informational RFC, February 1997.

[Kat97b] Y. Katsube, K. Nagami, H. Esaki, “Internetworking Based on Cell Switch Router - Architecture and Protocol Overview”, Proceedings of the IEEE, 85(12) pp 1998-2006, December 1997.

[Kra97] H. Krawczyk, M. Bellare, R. Canetti, “HMAC: Keyed-Hashing for Message Authentication”, RFC 2104, Feb. 1997

[Lan97] P. Lantz, “*Usage of H.323 on the Internet*”, Internet Draft, IETF, Work in progress, February 1997.

[Lau94] M. Laubach, ‘Classical IP and ARP over ATM’, IETF Standards Track RFC1577, January 1994.

[Lin97] S. Lin, N. McKeown, “*A Simulation Study of IP Switching*”, Proc. ACM SIGCOMM, Sep. 1997.

[Luc97] J. Luciani, D. Katz, D. Piscitello & B. Cole, ‘NBMA Next Hop Resolution Protocol (NHRP), IETF Internet Draft (work in progress), March 1997.

[Man97] A. Mankin, B. Braden, S. Bradner, M. O’Dell, A. Romanow, A. Weinrib, L. Zhang, ‘Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement Some Guidelines on Deployment’, IETF Standards Track RFC2208, September 1997.

[McC95] S. McCanne, and V. Jacobson, "Vic: A flexible framework for packet video", Proceedings of ACM multimedia, November 1995.

[Mik97] J. Mikkonen, J Aldis, G. Awater, A. Lunn, ‘The Magic WAND - Functional Overview’, To be published, 1997.

[Moy94] J. Moy, "Multicast extensions to OSPF", RFC 1584, March 1994.

[Nar96] T. Narten et al., RFC 1970, “Neighbor Discovery for IP Version 6”, August 1996.

[New96] P. Newman, G. Minshall and T. Lyon “*Flow Labelled IP: Connectionless ATM Under IP*” Ipsilon Networks Inc, IEEE Infocom 96, 1996.

[New96a] P. Newman, T. Lyon & G. Minshall, 'Flow Labelled IP: A Connectionless Approach to ATM', IEEE INFOCOM '96, March 1996.

[New96b] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon & G. Minshall, 'Ipsilon Flow Management Protocol Specification for IPv4 Version 1.0', IETF Informational RFC1953, May 1996.

[New96c] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon & G. Minshall, 'Transmission of Flow Labelled IPv4 on ATM Data Links', IETF Informational RFC1954, May 1996.

[New96d] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon & G. Minshall, 'Ipsilon's General Switch Management Protocol Specification Version 1.1', IETF Informational RFC1987, August 1996.

[New98] P Newman, G. Minshall, T. Lyon, "IP Switching - ATM Under IP", IEEE/ACM Transactions on Networking, 6(2), April 1998.

[Par95] G. Parulkar, D. Schmidt, J. Turner, "IP/ATM: A strategy for Integrating IP with ATM", ACM SIGCOMM, September 1995.

[Pau97] S. Paul, K. K. Sabnani, J. C.-H Lin, and S. Bhattacharyya, "Reliable Multicast Transport Protocol (RMTP)", IEEE JSAC, vol. 15, no. 3, April 1997.

[Per95] M. Perez, F. Liaw, A. Mankin, E. Hoffman, D. Grossman & A. Malis, 'ATM Signalling Support for IP over ATM', IETF Standards Track RFC 1755, February, 1995.

[Per96] C. Perkins, 'IP Mobility Support', IETF Standards Track RFC 2002, October 1996.

[Per97] M. Perez, 'ATM Signalling Support for IP over ATM - UNI 4.0 Update', IETF Internet Draft (Work in Progress), June 1996.

[Per97b] C. Perkins, D. Johnson, 'Route Optimisation in Mobile IP', IETF Internet Draft (work in progress), August 1997.

[Per98] R. Pereira, 'IP Payload Compression Using DEFLATE', Internet Draft (Work in Progress) <draft-ietf-ippcp-deflate-03.txt>, Feb 1998.

[Rat96] Rat web server: http://www-mice.cs.ucl.ac.uk/mice/rat/.

[Rek97] Y. Rekhter, B. Davie, D. Katz, E. Rosen, G. Swallow, "Cisco Systems' Tag Switching Architecture Overview", IETF Informational RFC 2105, February 1997.

[Ros98] E. Rosen, A Viswanathan, R. Callon, 'Multiprotocol Label Switching Architecture', Internet Draft (Work in Progress) <draft-ietf-mpls-arch-02.txt>, July 1998.

[Rek97b] Y Rekhter, B. Davie, E. Rosen, G. Swallow, D. Farinacci, D. Katz, "Tag Switching Architecture Overview", Proceedings of the IEEE, 85(12), pp1973-1983, December 1997.

[Sar94] H. Saran, & S. Keshav, "An Empirical Evaluation of Virtual Circuit Holding Times in IP-over-ATM Networks", Proceedings of IEEE Infocom, June 1994.

[Ses96] S. Seshan, H. Balakrishnan, & R. Katz, “Handoffs in Cellular Wireless Networks: The Daedalus Implementation and Experience”, Kluwer International Journal on Wireless Communication Systems, 1996.

[Sha98] A. Shacham, R. Monsour, R. Pereira, & M. Thomas, ‘IP Payload Compression Protocol (IPComp)’, Internet Draft (Work in Progress) <draft- ietf-ippcp-protocol-06.txt>, May 1998.

[She95] S Shenker, ‘Fundamental Design Issues for the Future Internet’, IEEE Journal on Selected Areas in Communication, 13(7), pp 1176-1188, September 1995.

[Tal97] A. Talukdar, B. Badrinath, & A. Acharya, “MRSVP: A Reservation Protocol for an Integrated Service Packet Network with Mobile Hosts, Technical Report, DCS-TR-337, Rutgers University, 1997.

[Ter98] A. Terzis, J. Krawczyk, J. Wroclaswki, & L. Zhang, ‘RSVP Operation over IP Tunnels’, IETF Internet Draft, August 1998.

[Tho98] S. Thomson and T. Narten, “IPv6 Stateless Address Autoconfiguration”, Internet draft, February 1998 (update from RFC 1971).

[Vat96] Vat web server: http://www-nrg.ee.lbl.gov/vat/.

[Wb96] Whiteboard software available through ftp://ftp.ee.lbl.gov/conferencing/wb/.

[Wah97] M. Wahl, T. Howes, and S. Kille, “Lightweight Directory Access Protocol (v3)”, RFC 2251, Dec. 1997.

[Wor97] T. Worster, A. Doria, “Levels of Aggregation in Flow Switching Networks”, Proceedings IEEE Electronics Industry Forum, May 1997, pp51-59.

[XTP95] XTP Forum, "Xpress Transport Protocol Version 4", Santa Barbara, CA, March 1995.

[Xyl97] G. Xylomenos and G. Polyzos, "IP multicasting for point-to-point local distribution", Proceedings of IEEE Infocom '97, 1997.