Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2007

NUNO JOÃO SILVA LOPES HENRIQUES SÉNICA

MECANISMOS DE MOBILIDADE RÁPIDA COM SUPORTE DE QdS

FAST-MOBILITY MECHANISMS WITH QOS SUPPORT

Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2007

NUNO JOÃO SILVA LOPES HENRIQUES SÉNICA

MECANISMOS DE MOBILIDADE RÁPIDA COM SUPORTE DE QdS

FAST-MOBILITY MECHANISMS WITH QOS SUPPORT

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Dr. Rui Aguiar, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e da Dra. Susana Sargento, Professora Auxiliar Convidada do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

dedicatória

Dedico este trabalho à minha namorada, pais e irmão pelo incansável apoio.

o júri

presidente

Doutor Joaquim Arnaldo Carvalho Martins
Professor Catedrático da Universidade de Aveiro

vogais

Doutor Mário Serafim dos Santos Nunes
Professor Associado com Agregação do Instituto Superior Técnico da Universidade Técnica de Lisboa

Doutor Rui Luís Andrade Aguiar
Professor Auxiliar da Universidade de Aveiro (Orientador)

Doutora Susana Isabel Barreto de Miranda Sargento
Professora Auxiliar Convidada da Universidade de Aveiro (Co-orientadora)

agradecimentos

Sinceros agradecimentos ao Instituto de Telecomunicações, aos colaboradores e professores do Grupo de Redes Heterogéneas do Instituto de Telecomunicações - Aveiro.
Especial agradecimento ao Dr. Telemaco Melia e ao Eng. Paulo Ferrer, ambos pertencentes aos Laboratórios de Redes da NEC Europe, pelo apoio e cooperação na realização deste trabalho.

palavras-chave

Mobilidade, Qualidade de Serviço, Protocolos de Mobilidade, MIP, HIP, FMIP, Cellular IP, Fast Handover, Simulação, Mobilidade Rápida, Integração, Avaliação.

resumo

A área das redes de comunicações está, neste momento, a deparar-se com um novo paradigma causado pela tendência de convergência de redes sem fios e celulares. Desta convergência resultará a existência de uma camada de rede integradora, para facilitar o suporte de mecanismos de Qualidade de Serviço e mobilidade. Aqui, o suporte de mobilidade rápida e transparente, sem ser perceptível pelo utilizador, tem sido alvo de muita atenção, apesar de ainda existirem algumas limitações no seu suporte. A mobilidade transparente entre redes celulares, sem fios e fixas, é ambicionada mas ainda não foi alcançada.

O trabalho realizado nesta Dissertação consiste na descrição, especificação, implementação e teste de uma nova arquitectura de mobilidade sobre o protocolo IP. Esta arquitectura é baseada no protocolo de mobilidade *Mobility Support for IPv6* e em extensões de *Fast Handovers for Mobile IPv6*, sendo capaz de efectuar *handovers* iniciados pelo terminal e pela rede. A mobilidade é transparente entre tecnologias de acesso heterogéneas, através da integração de mecanismos de qualidade de serviço, tais como autorização de *handovers*, controlo de acesso, reserva e atribuição de recursos no novo ponto de ligação e também integrada com subsistemas de autenticação. São também propostos outros mecanismos de mobilidade rápida que fazem uso do protocolo *multicast* para distribuir os fluxos de tráfego direccionados ao terminal, pelos *routers* de acesso vizinhos, permitindo que os terminais móveis mudem para qualquer *router* de acesso na vizinhança sem interrupção dos serviços em curso. Estes mecanismos foram projectados para terminais móveis com grandes requisitos de mobilidade.

No âmbito do projecto IST Daidalos foi efectuada a integração de uma rede de próxima geração (4G) de forma a permitir a realização de testes de desempenho e conformidade aos mecanismos propostos. A presente Dissertação efectua uma avaliação de desempenho de uma arquitectura de mobilidade, em cenários intra- e inter-tecnologia, numa rede de testes real. Nesta avaliação foram utilizadas as métricas de atraso, *jitter* e perdas de pacotes nas fases de preparação e execução do *handover*. O impacto deste processo em comunicações de dados sobre TCP e UDP é também analisado. A arquitectura e os resultados obtidos no demonstrador real são apresentados e discutidos.

keywords

Mobility, QoS, Quality of Service, Mobility Protocols, MIP, HIP, FMIP, Cellular IP, Fast Handover, Simulation, Fast Mobility, Integration, Evaluation.

abstract

The field of network communications is, nowadays, facing a new paradigm caused by the forthcoming convergence of cellular and wireless data networks, which seems unavoidable. This convergence will result on an integration layer, to ease the support for Quality of Service and mobility mechanisms. Here, the support for fast and seamless mobility, not perceptible by the user, has been getting much attention, although several limitations still exist in this support. Seamless mobility between cellular, wireless and wired data networks is envisioned, but not yet achieved.

The work performed in the scope of this thesis aims to describe, specify, implement and test a novel mobility architecture based on the IP protocol. This architecture is based on the mobility protocol *Mobility Support for IPv6* and on extensions of *Fast Handovers for Mobile IPv6* RFCs, and is able to provide mobile terminal and network initiated handovers. The mobility is seamless across heterogeneous access technologies, by integrating Quality of Service mechanisms, such as handover authorization, access control, resources reservation and allocation at the new point of attachment, also integrated with an authentication sub-system. Other novel fast mobility mechanisms are also proposed, which make use of the multicast protocol to distribute the traffic flows directed to the terminal during the handover process among the neighbour access routers, allowing the terminal to handover to any access router in the vicinity without disruption of the ongoing services. These latter mechanisms were designed to mobile terminals with high mobility requirements.

In the scope of the IST Daidalos framework an integration process of a next generation (4G) network was carried out in order to perform performance and compliance tests to the proposed mechanisms. Furthermore, this thesis also evaluates the performance of a mobility architecture, both in intra and inter-technology scenarios, in a real testbed. In this evaluation were considered metrics such as packet delay, jitter and loss of the handover in its preparation and execution phases. The impact of the handover on ongoing TCP and UDP data communications is also addressed. The architecture and results obtained from the real demonstrator are also presented and discussed.

Index

Index Of Figures

Index Of Tables

Acronyms

2G	2nd Generation Mobile Networks
3G	3rd Generation Mobile Networks
4G	4th Generation Mobile Networks
A4C	Authentication, Authorization, Accounting, Auditing and Charging
AAA	Authentication, Authorization and Accounting
AAAC	Authentication, Authorization, Accounting and Charging
ACK	Acknowledgement
AM	Aggregation Module
AN	Access Network
ANQoSB	Access Network QoS Broker
AP	Access Point
AR	Access Router
ARM	Advanced Router Mechanisms
BSC	Base Station Controllers
BU	Binding Update
CAN	Content Adaptation Node
CAR	Current Access Router
CARD	Candidate Access Router Discovery
CMS	Central Monitoring System
CN	Correspondent Node

CNQoSB	Core Network QoS Broker
CoA	Care of Address
COPS	Common Open Policy Service
CoT	Care of Test
CoTI	Care of Test Init
CT	Context Transfer
CTP	Context Transfer Protocol
DiffServ	Differentiated Services
DIO	Duplicat Information Object
DM/D&M	Duplicate and Merging
DNS	Domain Name Service
DVB	Digital Video Broadcast
DVB-H	DVB Handhelds
DVB-S	DVB Satellite
DVB-T	DVB Terrestrial
ESP	Encapsulating Security Payload
FBack	Fast Binding Update Acknowledgement
FBU	Fast Binding Update
FDDI	Fiber Distributed Data Interface
FDM	Frequency Division Multiplexing
FE	Functional Entity
FHO	Fast Handover
FMIPv6	Fast Mobile IPv6
FNA	Fast Neighbour Adverstisement
GNED	Graphical Network Editor
GRAAL	Generic Radio Access Adaptation Layer
GUI	Graphical User Interface
HA	Home Agent
Hack	Handover Acknowledgement

HI	Handover Initiate or Host ID
HIP	Host Identity Protocol
HIT	Host Identity Tag
HMIPv6	Hierarchical Mobile IPv6
HO	Handover
HoA	Home Address
HoT	Home Test
HoTI	Home Test Init
IAL	Interface Abstraction Layer
ICMPv6	Internet Control Message Protocol version 6
ID	Identifier
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IIS	Intelligent Interface Selection
IIS_ALG	IIS Algorithm Module
IIS_COSTS	IIS Costs Module
IntServ	Integrated Services
IP	Internet Protocol
IPSec	Secure IP
IPv6	Internet Protocol version 6
IRTF	Internet Research Task Force
KDC	Key Distribution Centre
L2	TCP/IP Model Layer 2
L2ID	Layer 2 Identifier
LAN	Local Area Network
LAPB	Link Access Protocol, Balanced
LLA	Link Layer Address
MAC	Medium Access Control Address

MAP	Mobility Anchor Point
MGEN	Multi Generator
MIHO	Mobile terminal Initiated HandOver
MIP	Mobile IP
MIPv6	Mobile IPv6
MM	Measurement Module
MMSP	MultiMedia Service Proxy
MMSPP	MultiMedia Service Provisioning Platform
MPEG	Moving Picture Experts Group
MPLS	Multi Protocol Label Switching
MSC	Mobility Switch Center
MT	Mobile Terminal
MTC	Mobile Terminal Controller
NAR	Next Access Router
nAR	New Access Router
NCoA	New CoA
NED	Network Description
NFS	Network File System
NIHO	Network Initiated HandOver
OSI	Open System Interconnection
PA	Performance Attendant
PAN	Personal Area Network
PANA	Protocol for carrying Authentication for Network Access
PAR	Previous Access Router
PBDE	Policy Based Decision Enforcement
PBNMS	Policy Based Network Management System
PCoA	Previous or Physical CoA
PDP	Policy Decision Point
PEP	Policy Enforcement Point

PIM	Protocol Independent Multicast
PM	Performance Manager
PrRtAdv	Proxy for Router Advertisement
QoS	Quality of Service
QoSAL	QoS Abstraction Layer
QoSB	QoS Broker
QoSC	QoS Client
RAL	Radio Access Layer
RAL-WLAN	Radio Access Layer for Wireless LAN technology
REA	Readdress Packet
REPL	Reply
REQ	Request
RFC	Request for Comments
RM	Registration Module
RSVP	Resource Reservation Protocol
RtSolPr	Router Solicitation for Proxy
RVS	RendezVous Server
SA	Security Association
SEND	Secure Neighbour Discovery
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SNR	Signal to Noise Ratio
SPI	Security Parameter Index
SPP	Service Provisioning Platform
SSID	Service Set Identifier
TAR	Target Access Router
TCP	Transmission Control Protocol
TD-CDMA	Time Division Code Division Multiple Access
TDM	Time Division Multiplexing

TLV	Type Length Value
TM	Terminal Mobility
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication System
VCoA	Virtual CoA
VID	Virtual Identifier
VoIP	Voice over IP
W-CDMA	Wideband Code Division Multiple Access
WLAN	Wireless Local Area Network

Chapter 1 Introduction

1.1 Motivation

With the parallel deployment of circuit switched and packet switched networks, which evolved towards voice traffic, vertical handovers, (namely, the change of point of connection to the infrastructure, when the wireless media change its type), have become a challenging research topic, especially considering market trends. For instance, mobile operators are announcing dual network handsets, which may be used as a standard handset, as well as an in-house VoIP phone via a company's internal wireless LAN network. Dual-network solutions do not currently allow users to keep ongoing calls while performing vertical handovers. Moreover, in envisioning environments with millions of customers (where the number of handovers is very large), scalable support of fast mobility becomes a strong requirement for these vertical handovers.

The vital role of the Internet Protocol version 6 (IPv6) [RFC2460] in new wireless systems for both voice and data traffic is well-known, offering a convergence layer for seamless mobility and Quality of Service (QoS) across heterogeneous networks, world-wide. The 3G standard [3gpp04] for wireless communications combines high-speed *mobile* access with IP-based services. In the emerging wireless world, next generation all-IP networks will aim at offering a broad range of services accessible anytime and anywhere regardless of the wireless/wired technology.

The IPv6 protocol has a fundamental role in heterogeneous environments, allowing the support of mobility, Quality of Service (QoS) and multicast IP. IPv6 already has some support for mobility, but in order to support efficient and transparent fast-mobility, the mobile user cannot notice any degradation or interruption of the ongoing service, neither the users he is communicating with. With the quick increase of users in wireless networks, scalability of mobility protocols is also an important factor, since a high number of handovers can occur simultaneously. Thus, multiple IETF protocols have been proposed in this framework.

One of the major challenges is then to propose solutions enabling fast intra and inter-technology mobility taking into account scalability and deployment issues. Note that different access technologies (e.g. 802.11, TD-CDMA) have different handover concepts: some are network initiated, others are mobile terminal initiated. A handover protocol for truly heterogeneous environments will need to support both cases.

While IP mobility IETF [RFC3775] protocols have been proved to work properly independently, due to their own nature, this is not sufficient for these integrated environments. In the framework of the increasing complexity of next generation integrated systems, those protocols are further required to operate in such a manner that they optimize resource consumption (e.g. bandwidth in wireless environments) and scale as much as possible (e.g. increasing number of customers, increasing network heterogeneity). These integration requirements may imply the development of improvements to these protocols. Specifically, this work considers the combination of (i) fast mobility issues in heterogeneous environments and (ii) QoS within a *Diffserv* based mechanism.

1.2 Objectives

The objective of this Master Thesis is to specify, implement and evaluate IPv6 Fast-Mobility architectures with the goal to provide transparent mechanisms of fast mobility. The proposals are specified targeting Operators Networks with integration of mobility and Quality of Service. The use of a multicast network will also be evaluated in order to decrease the packet loss and delays in communication within the handover time. This thesis thus aimed to:

- Study of Fast-Mobility mechanisms available and evaluate its weaknesses.
- Specification of a Fast-Mobility solution integrated with QoS requirements, supporting handovers initiated by the mobile terminal (taking into account mobility and user requirements) or by the network (taking into account network resources optimization).
- Integration of this solution with mechanisms for access router discovery (Candidate Access Router Discovery - CARD) available on access routers, intelligent selection of interfaces (Intelligent Interface Selection - IIS) on the mobile terminal and network performance evaluation entities.
- Specification of a Fast-Mobility solution supported by a multicast network to convey packets to the neighbours access routers within handover time, in order to decrease packet loss. Integration of QoS in this solution.
- Initial process of evaluation of the different proposed solutions by means of simulation.
- Implementation and evaltuation on a real testbed of one of the proposed solutions.

This work was also part of research activities at european and national level, in the scope of the IST-DAIDALOS project [daidalos] resulting on publications on an internal university magazine [senica05], an IRTF draft submission [melia05], a national conference [senica06] and on a special issue about Mobile and Ubiquitous Systems of *The Mediterranean Journal of Computers and Networks* [senica07].

1.3 Thesis Structure

This thesis is organized as follows: this chapter, makes a brief introduction and motivation of this Master Thesis and contains the objectives of the Master Thesis.

Chapter 2 describes the mobility protocols studied to support this work by making a brief analysis on their operation mode and introducing their possible interactions with Quality of Service subsystems and Multicast networks.

Chapter 3 presents the main architecture and modules developed in the scope of this thesis as well as the specification of protocols to achieve fast and seamless mobility in QoS environments, which were then integrated in the Daidalos network. The specification also included the use of multicast networks to support handovers and packet duplication.

A detailed specification of the messages format for the proposed protocol, description of the implementation of the protocol are presented in Chapter 4.

Chapter 5 then presents the implementation, evaluation work and description and analysis of results. as well as the work carried out to test its behaviour and performance, in terms of its impact in communications. Finally the limitations related to the implementation of the mechanisms are analyzed.

This work is then concluded in Chapter 6 which presents the final conclusions on this work and provides indications for future work.

Chapter 2 Protocols with Mobility Support

This chapter presents the studied mobility protocols and an introduction about mobility integration with QoS and Multicast.

2.1 Introduction

Current communications developments are strongly focused on design concepts and architectures for mobile environments, using similar techniques and interfaces across heterogeneous technologies. These include multiple types of access technologies, from home-networks and wireless LANs to campus wide wireless access, and from 2G/3G cellular networks to satellite networks. Current handsets are following these trends, now including dual-mode and multi-mode support, to permit connectivity across 2G/3G and WLAN-based or DVB networks.

Tomorrow’s mobile customers will expect the network (and in particular its technological structure, the so-called 4G networks) to “disappear” and be of no concern, providing seamless end-to-end services accessible anytime and anywhere across heterogeneous technologies. A strong requirement for this achievement is the ability to provide service continuity while moving, achieving seamless handovers across different cells and technologies.

Due to the extensive research on mobility protocols in the last years, several proposals for mobility protocols are addressed in the literature. The Mobile Internet Protocol version 6 (MIPv6) [RFC3775] is the current next generation IETF standard to provide global mobility management and to enable mobile terminals (MT) to roam across different networks, maintaining its reachability to and from other terminals in the Internet. Even if MIPv6 potentially enables mobile Internet users to be always reachable regardless of the specific access network technology, increasing multimedia demands from mobile users highlighted MIPv6 shortcomings. Real time audio/video applications underline the need to have in place mechanisms minimizing the large handover latency and service degradation (eg. packet loss) usually associated with MIPv6.

In next generation networks, fast mobility has to be considered along with QoS and authentication profiles. Context transfer is an additional technique to optimize the handover process, minimizing the information needed to be transferred between the mobile terminal and other entities involved. The idea behind this concept is to transfer the context information for a session including security context, policy, QoS (Diffserv or Intserv) and AAA data during the handover process, between the network entities.

Hierarchical Mobile IPv6 [RFC4140] and Cellular IPv6 [shelby00] offer fast and seamless local mobility. Other mobility mechanisms enhancing Mobile IPv6 to account for performance issues are further defined in the IETF, such as Fast Handovers [RFC4068]. The Fast Handover proposal, which represents the basis for this work, uses the ”make before break” approach, where the terminal signals its handover to the new network using its current connection through the old network. Moreover, during handover, the packets are sent to the mobile terminal both via the old and the new network to prevent the existence of packet loss.

Within mobility there are two paradigms in the terms of handover and lost of connectivity, notice that these paradigms are also present in both layer 2 and layer 3 mobility concepts:

- Make Before Break
- Break Before Make

The Make Before Break paradigm consists on the terminal to first create a new connection on the new point of attachment with the same characteristics it has on the old one. The handover is performed after everything is set up. When the mobile terminal arrives at the new Access Point (AP), the connectivity is maintained exactly as it was on the previous one. In this case, the handover is made before the connection is broken.

Break Before Make happens when the mobile terminal is not capable of predicting the handover and only understands the need to handover when the link is down. After reconnecting, it has to register as a new device loosing all ongoing connections. In this case, it is said that the connectivity breaks before the mobile terminal makes the handover.

Additionally each handover can be either started by the MT, called Mobile terminal Initiated Handover (MIHO), or by some decision point in the network that instructs the MT to move to a previously selected AP/AR, called Network Initiated Handover (NIHO).

In MIHO the terminal decides which AP/AR to connect to and may request to the network an authorization to perform the handover or simply start the process of disassociation and association to the new AP/AR.

In many situations, MTs do not have all the information required for an optimal handover decision. For instance, APs have more transmit power than terminals causing loss of connectivity before the terminal realizes it; by monitoring the signal strength reaching the APs (from the MTs), it is possible for the AP to predict a loss of connectivity before it occurs. Another example is an excess of terminals attached to an AP, that can cause degradation on the link quality for all; or maybe a MT starting a demanding application may leave no bandwidth for others. These examples show why NIHO support is essential and should be enforced by the network.

This chapter presents some of the best well known mobility approaches in the literature. It also presents some proposals for integration of mobility in heterogeneous networks with 4G requirements, as QoS and AAA, and addresses the integration of mobility and multicast. This chapter is organized as follows. Section 2.2 presents a description of some mobility protocols that were important to define the basis of the work in this Thesis. Next, an overview of mobility integration with QoS and Multicast will be presented, respectively in sections 2.3 and 2.4. Finally, section 0 presents a summary of this chapter.

2.2 Mobility Protocols

This section shows an overview on the mobility protocols and mechanisms considered. First, it is described mobility at layer two, considering technology dependent approaches for mobility. Then, Layer 3 mobility mechanisms are presented, such as Mobile IPv6, Fast Mobility for Mobile IPv6 and Hierarchical Mobile IPv6, followed by a brief description of other mobility protocols, such as Cellular IPv6 and Host Identity Protocol.

2.2.1 Layer 2 Mobility

Layer 2 mobility means that a mobile terminal is capable of changing its point of attachment at data layer. For instance, in WLAN, when a device is attached to an AP, it is able to perform a change in its point of attachment mantaining L2 connectivity, which will allow upper layers to keep communicating.

Whenever IP address changes are not involved, i.e. no movement between access routers, the mobility is performed at layer 2. It means that, when the mobile terminal is moving inside the same IP subnet, the mobility is supported by technology features, and data transmission is minimally affected while moving.

2.2.1.1 Mobility mechanisms

Several Layer 2 technologies already contain intrinsic mobility mechanisms to deal with mobility. Due to technology specificities, mobility mechanisms are not the same across technologies, preventing layers above of having the same behaviour independently of the technology used.

The IEEE 802.11 technology [802.11] has basic mobility support with no optimization regarding lost frames. In this case, prior to movement, the attached terminal must decide to move. This decision can be triggered by several factors such as signal strength, frame acknowledgement, missed beacons, etc. Afterwards, it needs to decide where to move and this can be achieved in two ways: after or before the decision, pre-emptive AP discovery or runtime AP discovery. After this decision is taken, the terminal moves to the new AP by issuing re-association frames to associate to the new AP. After this step, the terminal can resume the temporarily suspended session.

UMTS uses two different types of modulations, frequency division multiplexing (FDM) and time division multiplexing (TDM) [umts]. While TDM does not have any kind of support for handover, only providing hard handovers (disconnect from the current attachment point and connect to the new), FDM provides mechanisms for soft handovers, which means that the technology is able to connect to a new attachment point before disconnecting from the old one. Together with Layer 3 operations it is possible to provide fast and seamless handovers without service degradation.

2.2.2 Mobile IPv6

Regarding inter technology handovers and inter IP subnet mobility, layer 2 mechanisms have no support for enhanced handovers. This will cause a complete disruption for all the above layers. Layer 3 mobility mechanisms have to be enhanced to provide fast and seamless mobility, turning this process completely transparent to underlaying technologies. However, optimizations should be done whenever possible in order to use technology dependent mechanisms to improve layer 3 handovers.

When an IPv6 terminal changes its location, its IPv6 address might also change in order to maintain connectivity. Stateful and stateless address auto configuration for IPv6 are mechanisms to allow changing addresses when moving to a different network. However, when changing networks, the address also changes, the existing connections of the mobile terminal using the address assigned from the previously connected network cannot be maintained and are terminated.

Mobile IPv6 [RFC3775] allows an IPv6 capable terminal to change its location on an IPv6 network maintaining connectivity with other terminals. The major benefit of Mobile IPv6 is thus that, if the mobile terminal changes its location and addresses, the ongoing connections in the mobile terminal are still maintained. To grant this, the mobile terminal uses an always reachable specific address, called Home Address, which is used in every connection. MIPv6 creates a new care-of-address (CoA) that represents the mobile terminal's new location and advertises this to its correspondent nodes and to a mobility manager (Home Agent, HA) in the home network. To support mobile Internet users, a mobile terminal has thus two IP addresses assigned, one fixed (the ”identification” home address), and the other changing (the ”topologically correct” CoA).

Mobile IPv6 provides transport layer connection survivability when a mobile terminal moves from one network to another by performing address maintenance for mobile terminals at the IP layer.

2.2.2.1 Mobile IPv6 Components

Figure 1 shows the elements of Mobile IPv6.

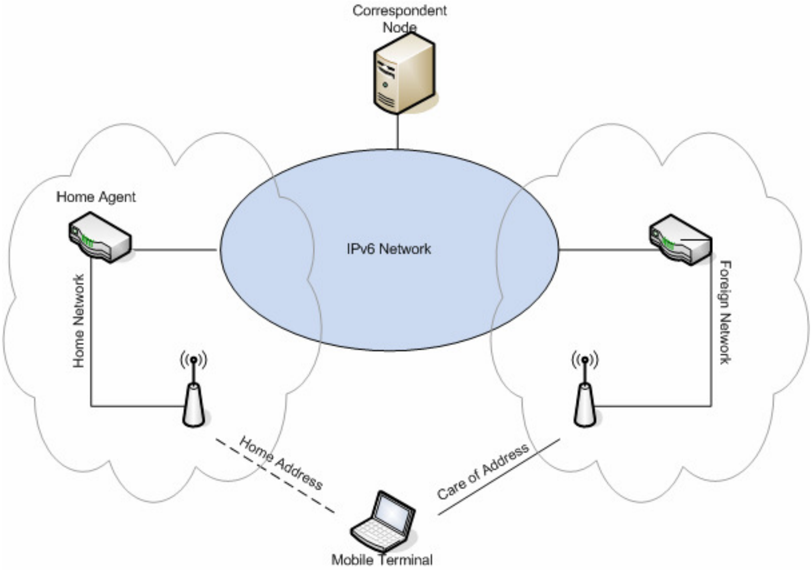


Figure 1 - Components of Mobile IPv6

Mobile IPv6 consists on the following components:

2.2.2.1.1 Home network

The home network refers to the network where it is assigned the home subnet prefix, from which the mobile terminal obtains its home address (HoA). The home agent resides on the home network.

2.2.2.1.2 Home address

The HoA is an address assigned to the mobile terminal when it is at the home network making it always reachable, regardless of its location on an IPv6 network. When the mobile terminal is attached to the home network, Mobile IPv6 procedures are not used and communication happens normally. If the mobile terminal is away from home (not attached to the home network), packets addressed to the mobile terminal's home address are intercepted by the HA and tunneled to the mobile terminal's current location on an IPv6 network. Because the mobile terminal is always assigned the HoA, it is always logically connected to the home network.

2.2.2.1.3 Home agent

The HA is a router on the home network that keeps registrations of mobile terminals that are connected to a foreign network, and the different addresses that they are currently using. If the mobile terminal is away from home, it registers its current address with the HA, which tunnels data sent to the mobile terminal's HoA to the mobile terminal's current address on an IPv6 network and forwards tunneled data sent by the mobile terminal – when there's no support for Route Optimization otherwise the traffic flows directly between CN and MT.

2.2.2.1.4 Mobile Terminal

The MT is an IPv6 terminal capable of changing networks and addresses, maintaining reachability using its HoA. A MT is aware of its HoA and the global address for the network to which it is attached (known as the CoA), and it sends its HoA/CoA mapping to the HA and Mobile IPv6-capable terminals with which it is communicating.

2.2.2.1.5 Foreign network

Foreign network is any network that is not the mobile terminal's home network.

2.2.2.1.6 Care-of address

CoA is an address used by a mobile terminal while it is connected to a foreign network. For stateless address configuration, the CoA is a combination of the foreign subnet prefix and an interface ID determined by the mobile terminal. A mobile terminal can be assigned multiple CoAs; however, only one is registered as the primary CoA with the mobile terminal's HA. The association of a HoA with a CoA for a mobile terminal is known as a binding. Correspondent nodes and HA keep information about bindings in a binding cache.

2.2.2.1.7 Correspondent node

Correspondent node (CN) is an IPv6 node that communicates with a mobile terminal. A CN does not have to be Mobile IPv6-capable. If it is Mobile IPv6-capable, it can also be a mobile terminal that is away from home.

2.2.2.2 Mobile IPv6 Transport Layer Transparency

To achieve transport layer transparency for the HoA while the mobile terminal is assigned a CoA, Mobile IPv6-capable terminals works on the following way:

a) When a mobile terminal that is away from home sends data to a correspondent node, it sends the packets from its CoA and includes the mobile terminal's HoA in a Home Address option in a Destination Options extension header. When the correspondent node receives the packet, it logically replaces the source address of the packet (the CoA) with the HoA in the Home Address option.

b) When an Mobile IPv6-capable correspondent node sends data to a mobile terminal that is away from home, it sends the packets to the CoA and includes a Type 2 Routing extension header containing a single address, the mobile terminal's HoA. When the mobile terminal receives the packet, it processes the Type 2 Routing header and logically replaces the destination address of the packet (the CoA) with the HoA from the Type 2 Routing header.

c) If a correspondent node is not Mobile IPv6-capable or if the communication is starting, then packets sent between the correspondent node and the mobile terminal that is away from home are exchanged via the home agent. The correspondent node sends packets to the mobile terminal's HoA. These packets are intercepted by the home agent and tunneled to the mobile terminal's CoA. The mobile terminal tunnels packets destined for the correspondent node to the home agent, which forwards them to the correspondent node. This indirect method of delivery, known as bidirectional tunneling, although inefficient, allows communication with legacy nodes and correspondent nodes that are not Mobile IPv6-capable or at the start of any communication prior to knowing the MobileIPv6 capabilities of each endpoints.

2.2.2.3 Handover procedure

From the Mobile IPv6 view, a terminal is capable of establishing data communications if it is receiving ICMPv6 Router Advertisement messages periodically which represent the access router network. Prior to establish data communications, the mobile terminal must be registered to its Home Agent, if it is not at its home network. This registration informs the Home Agent about the location of the mobile terminal. This registration is performed by the Binding Update message, which is triggered by the Router Advertisement and after configuring the CoA and the routes. This message is sent by a mobile IPv6 node that is away from home to update another node with its new CoA. This option is used both to update the home agent with a new CoA (home registration binding update), and to update a Mobile IPv6-capable correspondent node with a binding that maps the HoA of the mobile terminal to its CoA (correspondent registration binding update). The Home Agent, upon reception of the Binding Update will update its binding cache and will answer with a Binding Acknowledgment, if no errors were found on the registration of the Binding Update, or with a Binding Error informing the terminal about an error (informing that the registration was not performed). The Binding Acknowledgment includes an indication of binding lifetime; it also includes an indication of how often the mobile terminal should send binding updates.

If the registration was successful, the mobile terminal is now able to use its HoA as the default and reachable address for every data communication.

When the terminal handovers to a new network, it must start this described procedure again and register again its new CoA at the home agent. To ensure that a mobile terminal is reachable both through its HoA and CoA, a mechanism called return routability is performed (this mechanism is described in the following section). Making use of this mechanism, when data is flowing between the mobile terminal and correspondent nodes, it is possible to understand which type of routing will be used to maintain connectivity between them, such as route optimization and indirect delivery (this is described in sub-section 2.2.2.5.1).

2.2.2.4 Return Routability procedure

The Return Routability procedure establishes a proof to the correspondent node that the mobile terminal is reachable at its HoA and its CoA and determines tokens that are used to derive a binding management key, which is used to calculate authorization data values for binding messages.

Figure 2 shows the Return Routability procedure.

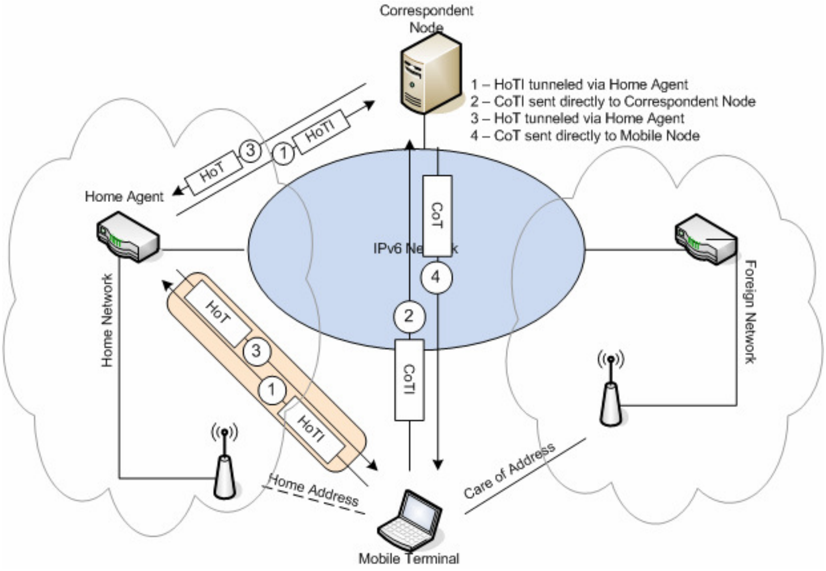

Correspondent Node
1 – HoTI tunneled via Home Agent
2 – CoTI sent directly to Correspondent Node
3 – HoT tunneled via Home Agent
4 – CoT sent directly to Mobile Node
Home Agent
HoT
3
1
HoTI
IPv6 Network
CoT
4
2
CoTI
Home Network
Foreign Network
Home Address
Care of Address
Mobile Terminal

Figure 2 - The Return Routability procedure

This procedure makes use of four messages:

- Home Test Init (HoTI) is sent by the mobile terminal to test the indirect path from a mobile terminal to a correspondent node via the home agent.
- Care-of Test Init (CoTI) is sent by the mobile terminal to test the direct path from a mobile terminal to a correspondent node.
- Home Test (HoT) is sent by the correspondent node to respond to the HoTI message.
- Care-of Test (CoT) is sent by the correspondent node to respond to the CoTI message.
- The full Return Routability process is the following:
- The mobile terminal sends a HoTI message indirectly to the correspondent node, tunneling the message through the home agent.
- The mobile terminal sends a CoTI message directly to the correspondent node.
- The correspondent node sends a HoT message in response to the HoTI message (sent indirectly to the mobile terminal via the home agent).
- The correspondent node sends a CoT message in response to the CoTI message (sent directly to the mobile terminal).

The correspondent node responds to the HoTI and CoTI messages as they arrive, independently of each other. The messages can arrive in any order. The correspondent node does not store any state information after responding to the HoTI or CoTI message.

The HoT message is sent to the mobile terminal's HoA. To provide security for the HoTI and HoT messages in the path from the home agent to the mobile terminal, the home agent can use Internet Protocol security (IPSec) Encapsulating Security Payload (ESP) tunnel mode to provide data confidentiality, data origin authentication, and data integrity for the HoT message. For information about how IPSec and ESP are used to protect Mobile IPv6 traffic, see [RFC3776].

From the tokens in the HoT and CoT messages, the mobile terminal can derive a binding management key. From information in the Binding Update message, the correspondent node can derive the same binding management key and use it to verify authentication data stored in the Binding Update message.

Note that the Return Routability procedure is designed to verify that the mobile terminal is reachable at both its HoA and CoA. The HoA must be verified to prevent spoofing of binding updates. The CoA must be verified to protect against denial-of-service attacks in which the correspondent node is tricked to flood a false CoA with packets.

2.2.2.5 Sending Data

Data between a mobile terminal that is away from home and a correspondent node can be sent in the following ways:

- Indirect delivery via Home Agent because there is no binding (bidirectional tunneling) at the CN.
- Direct delivery because there is a binding (route optimization) at the CN.

2.2.2.5.1 Indirect Delivery

When a correspondent node either does not yet have a binding for the mobile terminal (correspondent registration is in progress) or does not support Mobile IPv6, it sends packets to the mobile terminal using only its HoA. These data packets are forwarded to the HoA of the mobile terminal.

Figure 3 shows the correspondent node sending data packets to a mobile terminal that is away from home via indirect delivery.

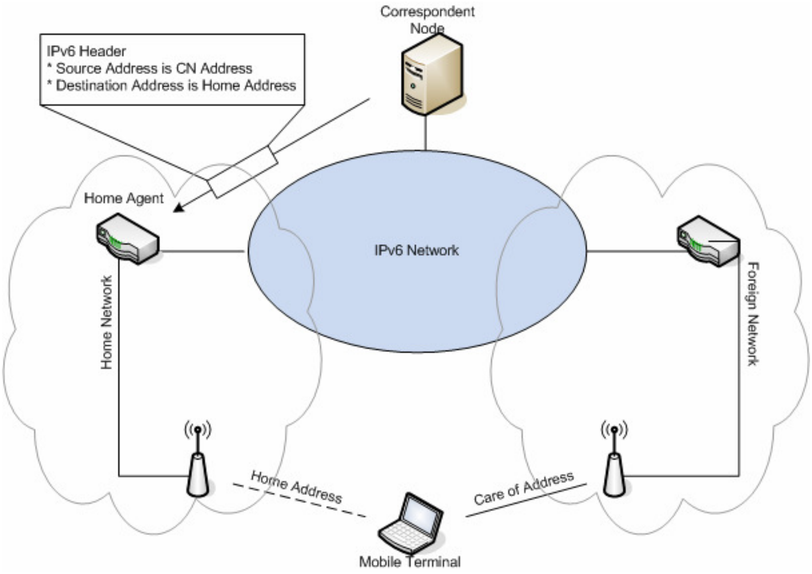


Figure 3 - Data packets sent by a correspondent node to the mobile terminal's HoA

When the home agent intercepts a packet sent to a mobile terminal's HoA, it tunnels the packet to the mobile terminal using the mechanism shown in Figure 4.

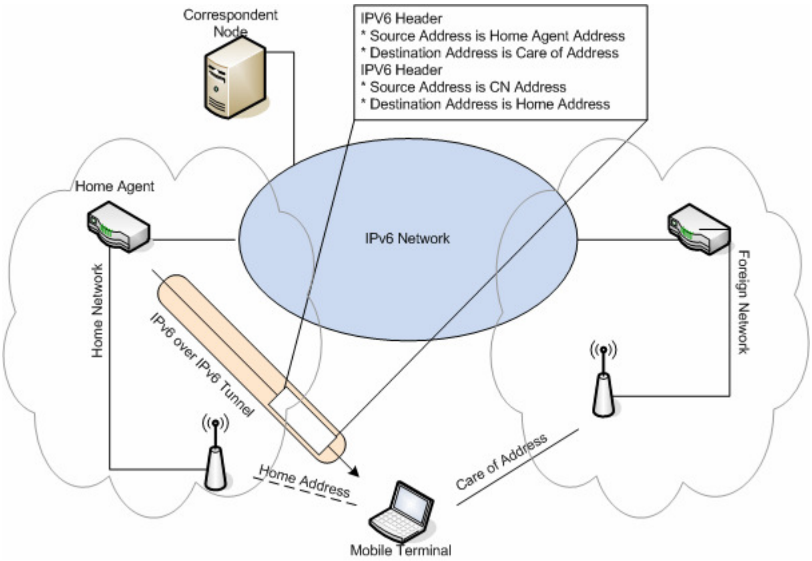


Figure 4 - Intercepted packet tunneled to a mobile terminal by its home agent

When anwering, the mobile terminal sends packets to the correspondent node by tunneling them to the home agent, as shown in Figure 5.

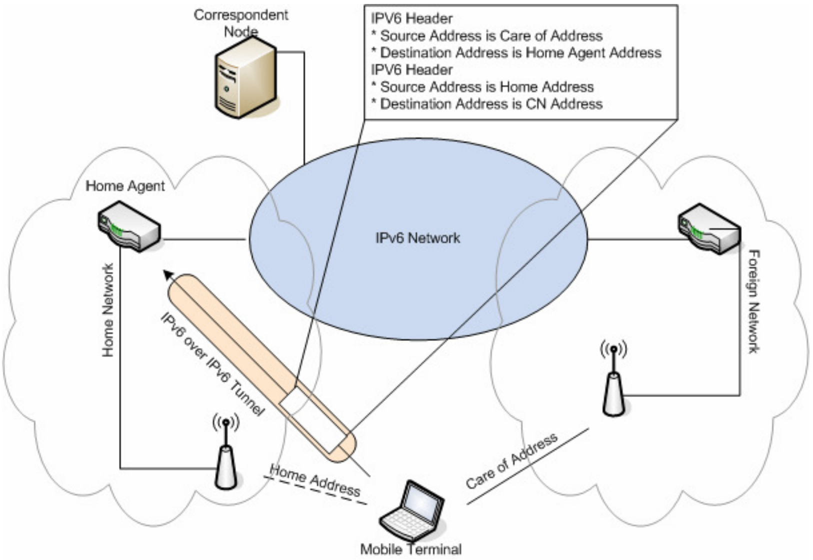


Figure 5 - Tunneled packets to a home agent

The home agent then forwards tunneled data packets from a mobile terminal to a correspondent node using the mechanism shown in Figure 6.

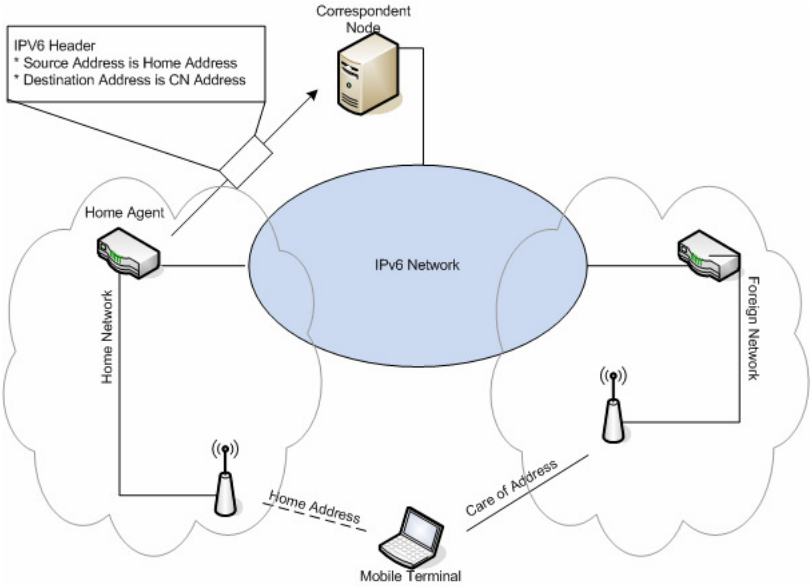


Figure 6 - Forwarded packet from a home agent to a correspondent node

2.2.2.5.2 Direct Delivery

When the mobile terminal is away from home, it can choose to either send data from its HoA using Mobile IPv6, or its CoA without using Mobile IPv6, based on the following. For transport layer connection data (such as TCP sessions) that are long-term, the mobile terminal sends the data from its HoA and includes the Home Address option. For short-term communication that does not require a logical connection, such as the exchange of Domain Name System (DNS) messages for DNS name resolution, the mobile terminal can send data from its CoA and not use a Home Address option. In this case, the mobile terminal is sending and receiving packets normally from its CoA.

Figure 7 shows data packets sent directly from the mobile terminal to the correspondent node when the mobile terminal has a binding update list entry for the correspondent node's address.

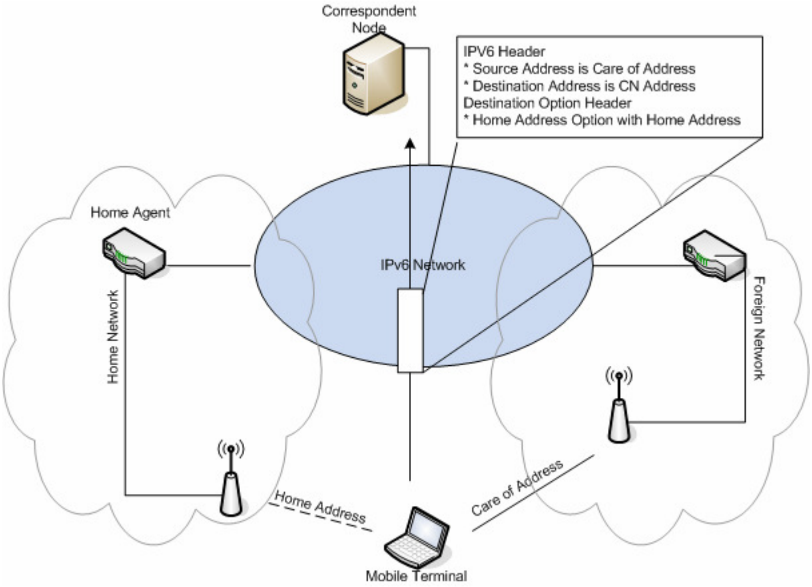

Correspondent Node
IPV6 Header
* Source Address is Care of Address
* Destination Address is CN Address
Destination Option Header
* Home Address Option with Home Address
Home Agent
IPv6 Network
Home Network
Foreign Network
Home Address
Care of Address
Mobile Terminal

Figure 7 – Packet sent from the mobile terminal to the correspondent node in direct delivery

2.2.3 Fast Handovers for Mobile IPv6

The Fast Handovers for Mobile IPv6 protocol (FMIPv6) [RFC4068] is a protocol designed to enhance the handover latency of Mobile IPv6 procedures, which aims at allowing a MT to configure a CoA prior to moving and connect at the new network. This allows the MT to use the new CoA as soon as it is attached to the new network. FMIPv6 provides a bidirectional tunnel between the new and the old access router which tries to eliminate latency while the Binding Update procedure is being performed.

Compared to MIPv6, the FMIPv6 protocol claims to be more efficient by eliminating IPv6 configuration by means of Router Discovery, Address Configuration and Duplicate Address Detection. The next sub-sections will detail this protocol.

2.2.3.1 Protocol Overview

The ability to immediately send packets from a new subnet link depends on the "IP connectivity" latency, which in turn depends on the movement detection latency and new CoA configuration latency.

Once a MT is IP-capable on the new subnet link, it can send a Binding Update to its Home Agent and one or more correspondents. Once its correspondents successfully process the Binding Update, which typically involves the Return Routability procedure, the MT can receive packets at the new CoA. In this case, the ability to receive packets from correspondents directly at its new CoA depends on the Binding Update latency as well as the IP connectivity latency.

Figure 8 shows the reference scenario for FMIPv6 handovers.

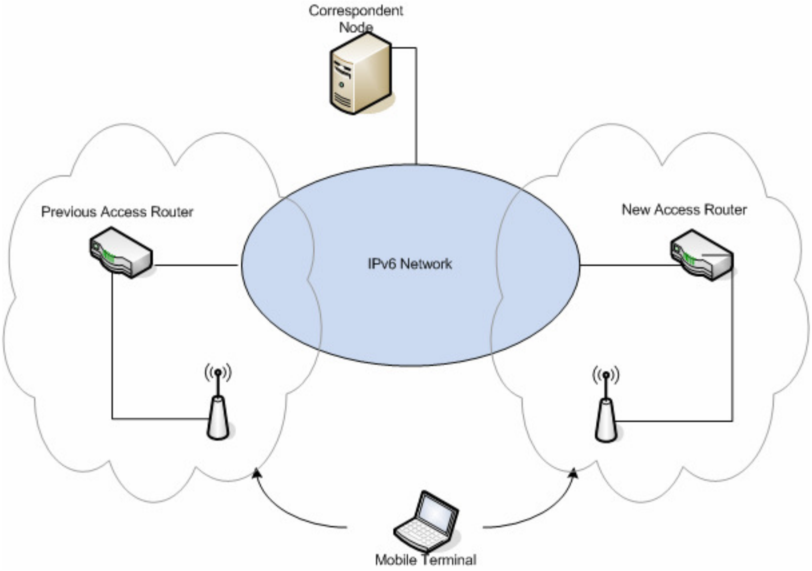


Figure 8 - Reference Scenario for Handover

FMIPv6 enables a MT to quickly detect that it has moved to a new subnet by providing the new access point and the associated subnet prefix information when the MT is still connected to its current subnet. For instance, a MT may discover available access points using link-layer specific mechanisms (i.e., a "scan" in WLAN) and then request subnet information corresponding to one or more of those discovered access points: it may do this after performing router discovery or at any time while connected to its current router.

FMIPv6 protocol requires that an access router has previous knowledge of surrounding access routers where the node is able to connect to. In addition, the knowledge of all L2 access points at each access router is also desired.

The protocol defines new terminology and new messages. In terms of terminology, it introduces the previous and new access routers (PAR and NAR), which are, respectively, the access routers that handle the terminal’s traffic and where the terminal is connected prior and after the movement, and the previous and new CoA (PCoA and NCoA, respectively), which are the addresses the MT uses before and after it moves.

The new messages introduced are the following:

- Router Solicitation for Proxy – RtSolPr (MT -> PAR). MT requests handover information to the PAR.
- Proxy Router Advertisements – PrRtAdv (PAR -> MT). PAR informs the MT of surrounding links.
- Fast Binding Update – FBU (MT -> PAR). MT performs the Binding Update with the NCoA included in the PrRtAdv message.
- Handover Initiate – HI (PAR -> NAR). PAR informs NAR about handover initiation
- Handover Acknowledgement – HAck (NAR -> PAR). NAR acknowledges the handover initiation.

- Fast Binding Acknowledgement – Fback (PAR -> MT). PAR acknowledges the FBU.
- Fast Neighbor Advertisement – FNA (MT -> NAR). MT announces its presence to the NAR.

2.2.3.1.1 Mobile Terminal Initiated Handover

An overview of the handover procedure is depicted in Figure 9. When the handover is initiated by the MT (which means that the MT takes the decision to move) it issues an RtSolPr message to its PAR in order to obtain information about its surrounding networks. In case of 802.11 networks, the RtSolPr message contains a list of access points that the MT can detect. The PAR replies with a PrRtAdv message which contains a list of IPv6 layer information for each AR associated to each AP that was identified in the previous message. This information includes link layer addresses of the ARs and the prefixes through which the MT can configure a new CoA.

After the reception of the PrRtAdv message, the MT can take the decision on which AP it is going to associate with. After this decision, it sends a FBU to the PAR with the information about the new AP and consequently the new AR. The PAR issues the HI message, which is acknowledged by the HAck message that verifies the correctness of the IPv6 configuration sent on the PrRtAdv message.

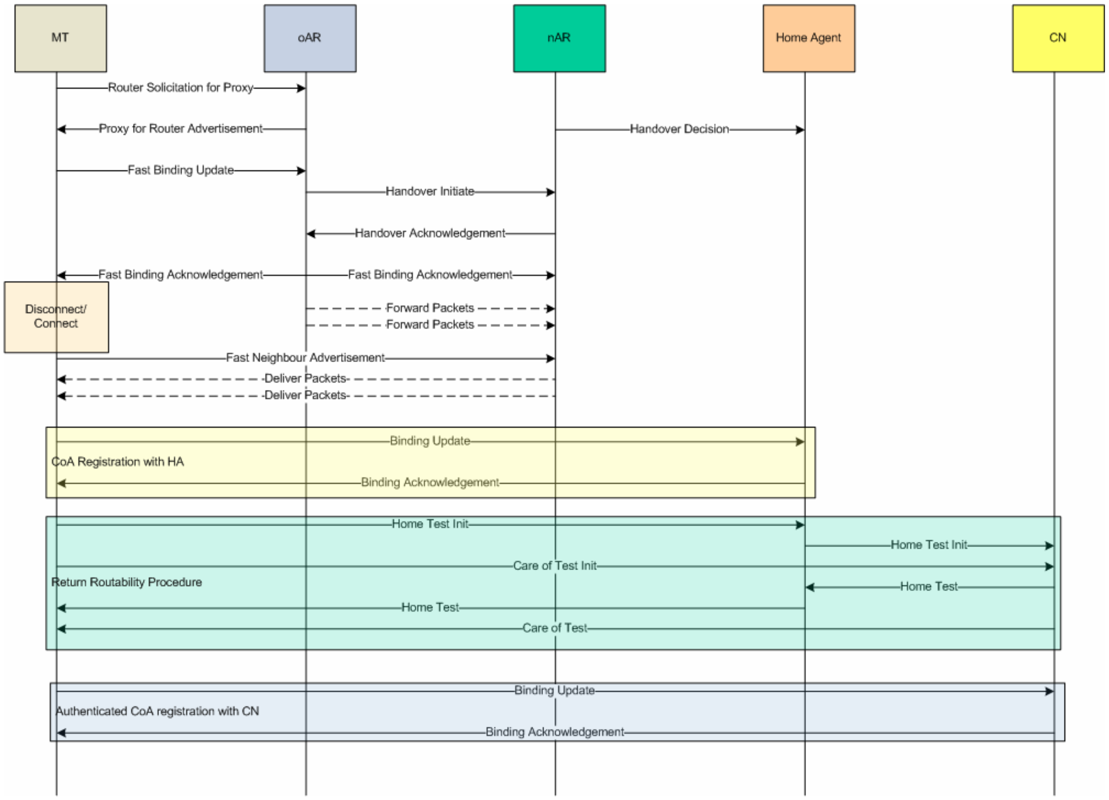


Figure 9 - FMIPv6 MT Initiated Handover Procedure

Upon the receipt of the HAck, the PAR establishes a binding between PCoA and NCoA and tunnels any packets previously destined to the PCoA, to the NCoA. The PAR will then forward packets to the NCoA via the bi-directional tunnel. The NAR buffers these packets until the MT arrives on its network and after association the PAR delivers the packets to the MT. The MT announces its presence on the new link by sending a FNA message. While the MIPv6 Binding Update procedure is not completed, the MT still uses the bi-directional tunnel. Notice that the regular MIPv6 handover procedure only occurs after the FMIPv6 tunnel, which performs binding update and registration with the Home Agent and the Correspondent Nodes.

In this way, packets that were normally lost during movement will be buffered by the NAR and delivered to the MT as soon as it arrives to the new link. Furthermore, communication with CNs can continue via the bi-directional tunnel. Although latency effects on the real-time traffic will continue to exist, they are reduced only to the time it takes to actually move (disconnect from PAR and connect to NAR).

2.2.3.1.2 Network Initiated Handover

In some network deployments, it may be possible for the network to initiate the handover procedure rather than the MT. As an example scenario an intelligent subsystem on the PAR can determine that a MT would be better served if connected to another router in the neighborhood (load balance scenarios). In such situations, the PAR sends an unsolicited PrRtAdv to the MT containing the information with which the MT can connect to the new network. Apart from the absence of the initial RtSolPr message, the messages exchange is the same as in Figure 9.

2.2.3.1.3 Reactive Handovers

So far it is assumed that the MT only moves to the new network once the FBU has been sent to the PAR. However, the situation where the MT moves to the new network before it had the chance to send the FBU can arise. In such a case, the MT sends the FBU encapsulated inside the FNA, which is then forward to the PAR allowing the PAR to make the PCoA-NCoA binding and forward any packets previously destined to the PCoA, to the NCoA. In this situation, the time between the MT moving and the reception of the FBU by the PAR can potentially affect the packet loss.

2.2.4 Hierarchical Mobile IPv6

Although Mobile IPv6 increased the efficiency of routing between MTs and the correspondent nodes, since packets are sent directly in both directions, binding updates are always sent to home agent and to all correspondent nodes communicating with the MT. When MTs are moving frequently between cells the signaling load introduced by the binding updates becomes significant.

The approach of Hierarchical Mobile IPv6 (HMIPv6) [RFC4140] is to distribute the management of the handoffs to reduce the amount of signaling in the wireless network; this also increases the efficiency of MIPv6 in terms of handoff speed. The next sub-sections will detail this protocol.

2.2.4.1 Hierarchy

Hierarchical Mobile IPv6 uses a new node denoted as Mobility Anchor Point (MAP). MAP is an entity which assists with MIPv6 handoffs. MAPs are deployed in a treelike structure as shown in Figure 10. Most of the handoffs will occur at the lowest level, and so, most of the signaling load is handled near the bottom of the hierarchy. The delay of the handoffs will decrease when they are handled closer to the mobile terminal.

The Hierarchical scheme introduce minor extensions to the MT and the HA operations, and it will not affect the correspondent node operation. The Highest MAP in the hierarchy will receive all the packets, coming from the outside of the hierarchical domain, on behalf of the MT and encapsulates and forward them to the MT's CoA within a local MAP domain. MAPs act as a local home agent for the MT.

The MT registers unique virtual CoA (VCoA) from the highest MAP to HA and CNs outside the hierarchy. This binding does not change when MT moves inside the hierarchy. In this sense, the movement of the MT inside the hierarchy remains invisible to the host outside the hierarchy. Also the existence of the hierarchy is invisible to the correspondent host outside the hierarchy. The MT has a unique VCoA from every MAP in the path from the root of the hierarchy to the lowest MAP. In addition to the VCoA, the MT also has a physical CoA (PCoA) which it uses when communicating with hosts in the same hierarchy.

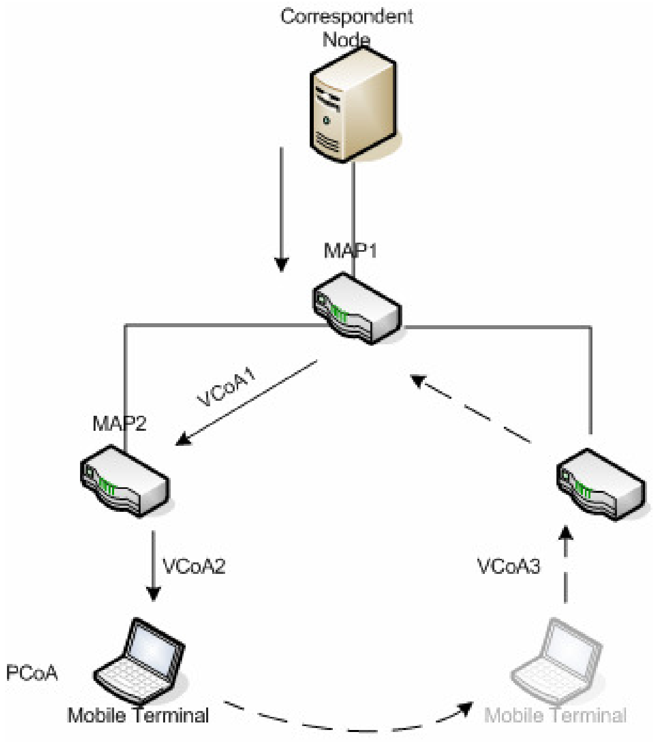


Figure 10 - Hiererchical foreign agents

In Figure 10 the hierarchical structured MAPs can offer seamless mobility to the mobile terminal when it moves from MAP2 to MAP3 while communicating with correspondent node. When the MT arrives in a foreign network, it registers the unique VCoA1 with the home agent and correspondent nodes. As previously referred, it also acquires a unique VCoA from each level of the hierarchy. In this sense, when the MT moves to MAP2, it acquires VCoa2 from MAP2 and VCoA1 from MAP 1. When MAP1 receives a packet addressed to VCoA1, it can determine the next VCoA by looking up the binding between VCoA and the next lower VCoA. After this process, MAP1 tunnels the packet to this VCoA. This continues until the datagram reaches the lowest MAP, which tunnels the packet to the MT's PCoA. In Figure 10, MAP1 tunnels packets to VCoA2, MAP2 decapsulates the packet, encapsulates it and tunnels it to MT's PCoA.

2.2.4.2 Registration procedure

This section provides an overview on the registration procedure used in HMIPv6.

2.2.4.2.1 Autoconfiguration of the care-of addresses

When a MT enters a new foreign sub network it first acquires a new PCoA by means of address autoconfiguration. PCoA must be globally routable, because MT uses it as the source address of all datagrams that it sends. MT forms its PCoA with the network prefix and the EU-64 bit string. If the MT tries to send a binding update with a duplicate address, the MAP replies to the MT with an error message, requesting it to try again with a random bit string as the host address. A similar mechanism can be used for creating the unique VCoAs.

2.2.4.2.2 Inter site handoff

When the MT receives a router advertisement with the mobility information option that contains a new hierarchy, it sends a binding update. This message binds its PCoA to its lowest VCoA, to the lowest MAP. After this process, the lowest MAP sends another binding update to the next higher MAP. These messages form a binding between the VCoAs of the MT in the MAPs. This continues until the highest MAP receives a binding update: it checks that the MT is allowed to use the network and sends a binding acknowledgement to the next lower MAP. The MAPs also store the security association with the MT. These acknowledgements are sent until the lowest MAP receives one. Then, the lowest MAP sends an acknowledgement to the MT. This procedure is shown in Figure 11. The MT sends a binding update to the MAP2, which is the lowest MAP: MAP2 sends a binding update to MAP1.

The MAPs create a preliminary entry of the binding into their binding caches. MAP1 is the highest MAP and it processes the authentication header of the original binding update and authenticates the MT. After successful authentication, it sends a positive binding acknowledgement to MAP2. The acknowledgement propagates downwards in the hierarchy and all the MAPs in the path update the status of the binding for the MT in their binding caches. When the MT receives the acknowledgement, it can start to use the foreign network and sends a binding update to its home agent and to all its correspondent nodes.

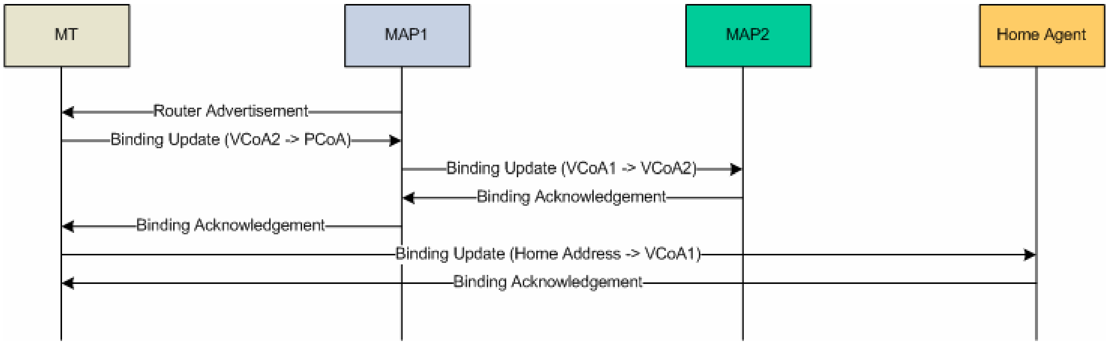


Figure 11 - The inter site hand-of procedure (HMIPv6)

2.2.4.2.3 Intra site handoff

When the MT acquires a new PCoA, it sends a binding update to the lowest MAP binding its VCoA to its PCoA. The lowest MAP sends the message to the second lowest MAP which binds the two VCoAs of the MT. This continues upwards on the path from the MT towards the root of the MAP tree until founding a MAP that already has a binding for the MT. That MAP is the lowest MAP that remains the same after the MTs handoff (it is also the last MAP that receives the binding update). The binding acknowledgements are sent in the same way as in the inter site handoff.

In Figure 12 the MT moves from MAP2 to MAP3. MT sends a binding update to MAP3, which is the lowest MAP: MAP3 sends a binding update to MAP1, which is the lowest MAP that has already a binding for the MT. MT also sends binding updates to local correspondent nodes that need to know its change of location. In this case, the MT does not send binding updates to home agents and correspondent nodes that are outside the hierarchy.

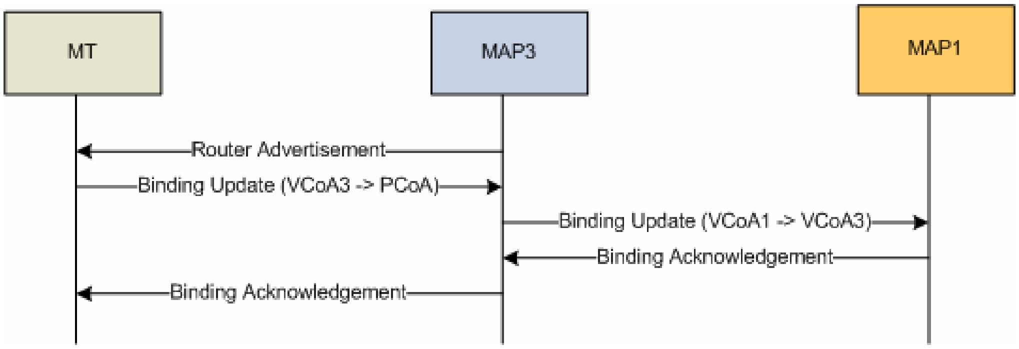


Figure 12 - The intra site hand-of procedure (HMIPv6)

2.2.5 Other Protocols

This sub-section presents other mobility protocols considered for this work which use different paradigms to achieve mobility.

2.2.5.1 Cellular IPv6

Although Mobile IPv6 is a powerful Internet mobility protocol, it presents some weaknesses for frequently migrating hosts. Specifically, after each host migration, a local temporary address must be obtained and communicated to a possibly distant Home Agent.

Cellular IPv6 [shelby00] is again a hierarchical approach, which combines the efficiency and scalability of IP with inherent features found in cellular networks, such as seamless handoff support, passive connectivity and paging. Thus, Cellular IPv6 is a Mobile IPv6 protocol extension and not a replacement.

A Cellular IPv6 network is comprised of a Gateway router that connects the network to the Internet as well as a set of nodes that are responsible for routing packets to MTs. The MTs are connected to the network via wireless access points denoted as Base Stations.

The main design issues for Cellular IPv6 are:

- The use of IPv6 extension headers to carry control information.
- Authentication transactions based on IPv6 authentication headers.
- Deployment of IPv6 stateless address autoconfiguration to obtain a CoA.
- The use of IPv6 CoA to identify MTs.

Mobile-IPv6-capable hosts use their IPv6 CoA as the source of every packet they send, and carry their permanent IPv6 HoA into a Home address destination option header. In order to be in line with Mobile IPv6 specification, the Cellular IPv6 control packets (route-updates and paging-updates) are sent uplink with MT CoA as source address.

On the reverse direction, IPv6 packets destined to a MT reach the Cellular IPv6 Gateway in two alternative structures, IPv6 encapsulated or in routing header.

When IPv6 encapsulation is used, the sender is not aware of the recipient MT's current CoA, and sends the packet destined to its HoA. This packet is normally routed to the MT's home network, where it is intercepted by the local Home Agent, which next encapsulates and sends the packet to the MT's current CoA.

When an IPv6 routing header is used the sender has a fresh binding for the recipient MT and sends the packet directly to its current CoA. In this case, the sender maps the MT's HoA as the last entry in the routing header (the MT's current CoA is mapped as second-to-last). The packets addressed to a MT will be routed towards the Cellular IPv6 Gateway/Router using prefix-based routing. Then, Cellular IPv6 host-based routing will forward packets to the MT, through the base station that it is currently attached to.

2.2.5.1.1 Features

Cellular IP inherits from cellular systems principles for mobility management (see Figure 13), passive connectivity and handoff control, but is designed based on the IP paradigm. The basic component of a Cellular IP network is the base station which serves as a wireless access point: it also routes IP packets and integrates cellular control functionality traditionally found in Mobile Switching Centers (MSC) and Base Station Controllers (BSC). The base stations are built on regular IP forwarding engines, but IP routing is replaced by Cellular IP routing and location management. The Cellular IP network is connected to the Internet via a gateway router. Mobility between gateways (i.e., Cellular IP access networks) is managed by Mobile IP, while mobility within access networks is handled by Cellular IP. The MTs attached to the network use the IP address of the gateway as their MIP CoA.

Assuming that there is no route optimization, packets will be first routed to the host's home agent and then tunneled to the gateway. The gateway "detunnels" packets and forwards them towards base stations. Inside the Cellular IP network, mobile hosts are identified by their HoA and data packets are routed without tunneling or address conversion. The Cellular IP routing protocol ensures that packets are delivered to the host's actual location. The packets transmitted by the MTs are first routed to the gateway and from there on to the Internet.

In Cellular IP, location management and handoff support are integrated with routing. To minimize control messaging, regular data packets transmitted by MTs are used to establish host location information. Uplink packets are routed from MTs to the gateway on a hop-by-hop basis. The path taken by these packets is cached in base stations. To route downlink packets addressed to a MT the path used by recent packets transmitted by the host is reversed. When the MT has no data to transmit, then it periodically sends empty IP packets to the gateway to maintain its downlink routing state. Following the principle of passive connectivity, MTs that have not received packets for a certain period of time allow their downlink soft-state routes to timeout and be cleared from the routing cache.

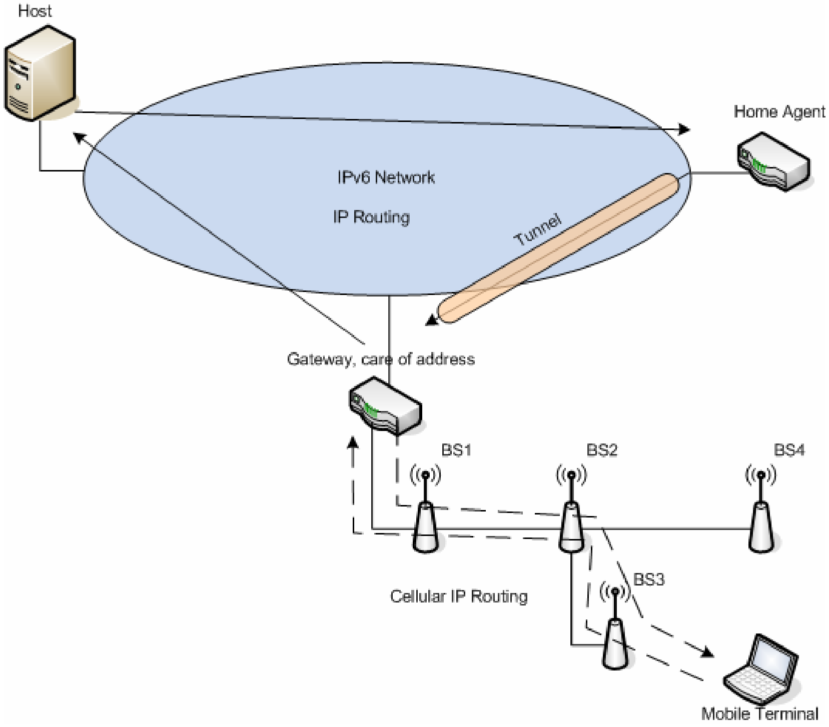


Figure 13 - Cellular IP Routing

2.2.5.1.2 Mobility Procedures

The Cellular IP hard handoff algorithm is based on a simplistic approach to mobility management that supports fast and simple handoff at the cost of potentially some packet loss. Handoff is initiated by MTs: they listen to beacons transmitted by base stations and initiate handoff based on signal strength measurements. To perform a handoff, a MT has to tune its radio to the new base station and send a route-update packet. This creates routing cache mappings on the route to the gateway, hence configuring the downlink route to the new base station. Handoff latency is the time that elapses between the handoff and the arrival of the first packet through the new route. For hard handoff, this equals the round-trip time between the mobile host and the cross-over point, which is the gateway in the worst case. During this time, downlink packets may be lost. The mappings associated with the old base station are not cleared at handoff, rather, they timeout as the associated soft-state timers expire.

Before the mappings timeout, a period exists when both the old and new downlink routes are valid and packets are delivered through both base stations. This feature is used in the Cellular IP semisoft handoff algorithm that improves handoff performance but still suits the lightweight nature of the base protocol, providing probabilistic guarantees instead of fully eliminating packet loss. Semisoft handoff adds one additional state variable to the existing mobile state maintained at mobile hosts and base stations. The semisoft handoff procedure has two components. First, in order to reduce handoff latency, the routing cache mappings associated with the new base station must be created before the actual handoff takes place. When the mobile host initiates a handoff, it sends a semisoft packet to the new base station and immediately returns to listening to the old base station. While the host is still in contact with the old base station, the semisoft packet configures routing cache mappings associated with the new base station. After a semisoft , the host can perform a regular handoff. The semisoft delay can be an arbitrary value between the mobile-gateway round-trip time and the route-timeout. The delay ensures that by the time the host tunes its radio to the new base station, its downlink packets are delivered through both the old and new base stations.

While the semisoft packet ensures that the mobile host continues to receive packets immediately after handoff, it does not, fully assure smooth handoff. Depending on the network topology and traffic conditions, the time to transmit packets from the cross-over point to the old and new base stations may be different, and the packet streams transmitted through the two base stations will typically not be synchronized at the mobile host. If the new base station "lags behind" the old base station, the mobile host may receive duplicate packets. However, the reception of duplicate packets in this case is not disruptive to many applications. If, otherwise, the new base station "gets ahead" then packets will be deemed to be missing from the data stream observed at the receiving mobile host.

The second component of the semisoft handoff procedure is based on the observation that perfect synchronization of the two streams is not necessary. The condition can be eliminated by temporarily introducing into the new path a constant delay sufficient to compensate, with high probability, the time difference between the two streams. This can be best achieved at the cross-over switch that understands that a semisoft handoff is in progress due to the fact that a semisoft packet has arrived from a mobile host that has a mapping to another interface. The mapping created by the semisoft packet has a flag to indicate that downlink packets routed by this mapping must pass a "delay device" before transmission. After handoff, the mobile host will send data or route-update packets along the new path, which will clear this flag and cause all packets in the delay device to be forwarded to the mobile host.

2.2.6 Host Identity Protocol

The current Internet uses two global name spaces: domain names and IP addresses. As we know, IP addresses have two uses: topological locators and network interface identifiers. The dual functionalities of IP addresses limit the flexibility of the Internet architecture such as IP address renumbering. In addition, transport layers are bound to IP addresses. This is becoming a serious problem for mobility and multi-homing.

The Host Identity Protocol (HIP) architecture [RFC4423] (see Figure 14) defines a new name space, the Host Identity name space, that decouples the name and locator roles currently filled by IP addresses. With HIP, transport layer operates on Host Identities instead of using IP addresses as endpoint names. At the same time, network layer uses IP addresses as pure locators.

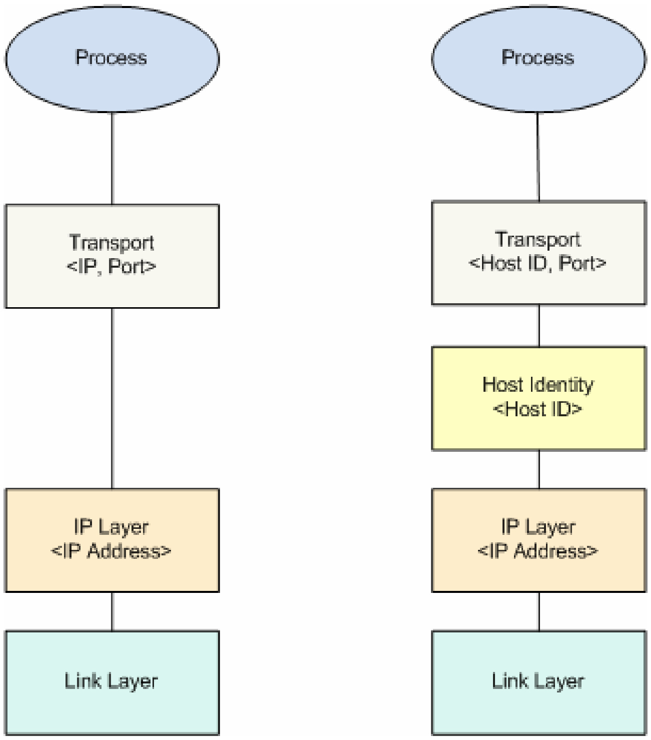


Figure 14 - HIP Stack

Each host has one or more asymmetric key pairs. The public part of this pair is used as Host Identifier (HI) that resides on the Host Identity layer. This layer was added between the transport and the network layer. The host itself is defined as the entity that holds the private part of the key pair. A hash of the HI, the Host Identity Tag (HIT), is used in HIP related protocols to represent the Host Identity. The HIT is 128 bits long and has the following three key properties:

- Has the same length of an IPv6 address and can be used in address sized fields in legacy APIs and protocols.
- It is self certifying (given a HIT it is computationally difficult to find the HI that matches the HIT).
- The probability of HIT collision between two hosts is very low.

At the transport layer. sockets are bound to HIs rather than IP addresses which are used to address and route packets just as today. The HIP protocol is composed of two round-trip processes (Figure 15), end-to-end Diffie-Hellman [dh] key exchange protocol, a mobility exchange and some additional messages.

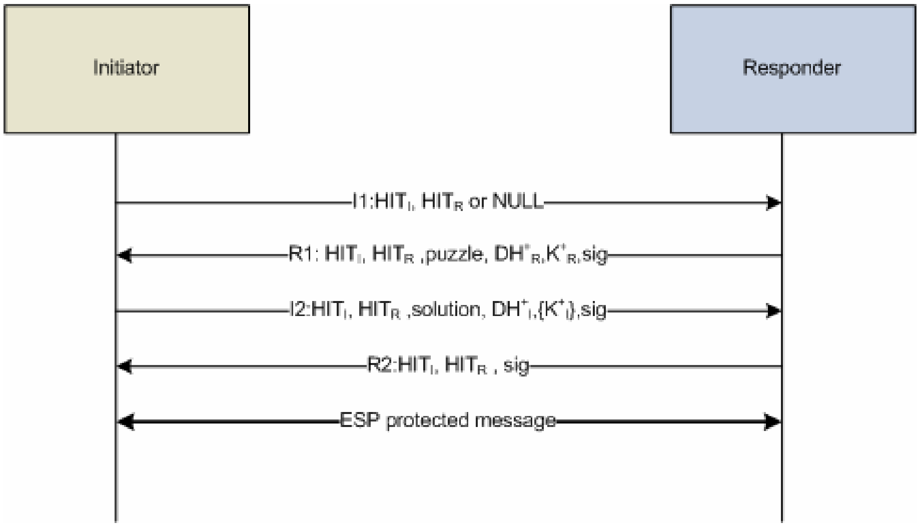

Initiator
Responder
I1:HIT$_I$, HIT$_R$ or NULL
R1: HIT$_I$, HIT$_R$,puzzle, DH^+_R,K^+_R,sig
I2:HIT$_I$, HIT$_R$,solution, DH^+_I,{K^+_I},sig
R2:HIT$_I$, HIT$_R$, sig
ESP protected message

Figure 15 – HIP base message exchange

The process for initiating a communication with a HIP enabled host is as follows. Whenever the initiator (*I*) wants to start a communication with another host (*R*), it starts to lookup *R*'s address at a domain name service. The DNS returns *R*'s address and HI/HIT parameters. At this point in time, *I* can initiate the communication and sends a message to *R* informing about its intentions to use HIP for that communication. To make sure that *I* can also handle HIP, *R* asks *I* to handle the HIP cookie sent in the response. *I* then computes and answers with the solution which is then confirmed by *R*. Hereafter, the data communication follows encrypted by ESP protected data security mechanism.

The above shows the process of base exchange. However, for a whole HIP process, the initiator looks up HI/HIT of responder from DNS or RVS (Rendezvous Server) [laganier06] firstly.

In the client side, the application sends DNS query to a DNS server. The DNS server replies with HI instead of IP address. In a second step, another lookup is made in the Host Identity layer by HIP daemon. This time, Host Identities are translated into IP addresses (HI->IP) for network layer delivery.

All this process will also create a security association based on IPSec bound to HI that will provide end to end security between both nodes: initiator and responder.

2.2.6.1 Mobility

[nikander04] proposes a generalization of an address called a "locator". A locator defines the point-of-attachment to the network. Using this parameter a mobile terminal is able to directly inform a correspondent node, with whom the host has an active HIP association, of a location change.

Since the Security Associations (SA) are not bound to IP addresses, the mobile terminal is able to receive packets that are protected using a HIP created Encapsulated Security Payload (ESP SA) from any address. Thus, a mobile terminal can change its IP address and continue to send packets to its peers.

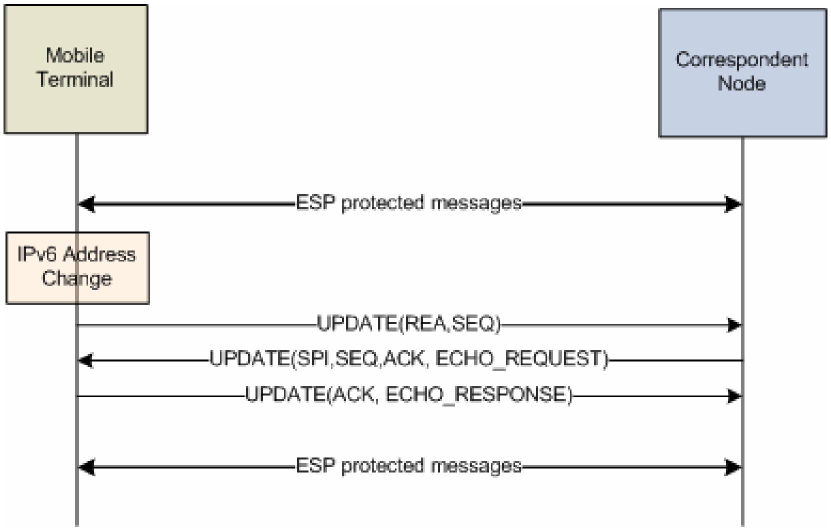


Figure 16 - HIP Mobility Process

Figure 16 depicts the mobility process. In the beginning, the MT is in address 1 and it moves to the address 2. During this process, the MT is disconnected from the peer host for a brief period of time while it switches from address 1 to address 2. Upon obtaining a new IP address, the MT sends the locator a readdress packet (REA) to the peer host in an update message. The REA indicates the following messages: the new IP address, the Security Parameter Index (SPI) associated with new IP address, the address lifetime and whether the new address is a preferred address. In Figure 16, the peer node performs an address check and solicits a response from the MT. Depending on whether the MT initiated a rekey, and on whether the peer host itself wants to rekey to verify the MT's new address, the process can be categorized into three cases: (1) readdress without rekeying, but with address check, like in Figure 16; (2) readdress with mobile-initiated rekey; and (3) readdress with peer-initiated rekey.

2.2.6.2 Multihoming

The locator concept used for mobility reasons also supports multihoming. A HIP node might have multiple locators simultaneously as in the case of mobility. It is possible for a node to send to a correspondent node several locator parameters at which it can be reached, and inform of which of those is the preferred one.

A node may sometimes have more than one interface. The node may notify the peer node of the additional interfaces. As an example, consider a multihoming node with two IP addresses, addr1 and addr2. If addr 1 is the preferred address the multihoming node sends update packets including addr1 and addr2 to its peer node. The peer node sends update packets to each address and updates corresponding SPIs.

Figure 17 depicts the multihoming procedure performed by HIP in order to inform correspondent nodes about the existence of more than one interface.

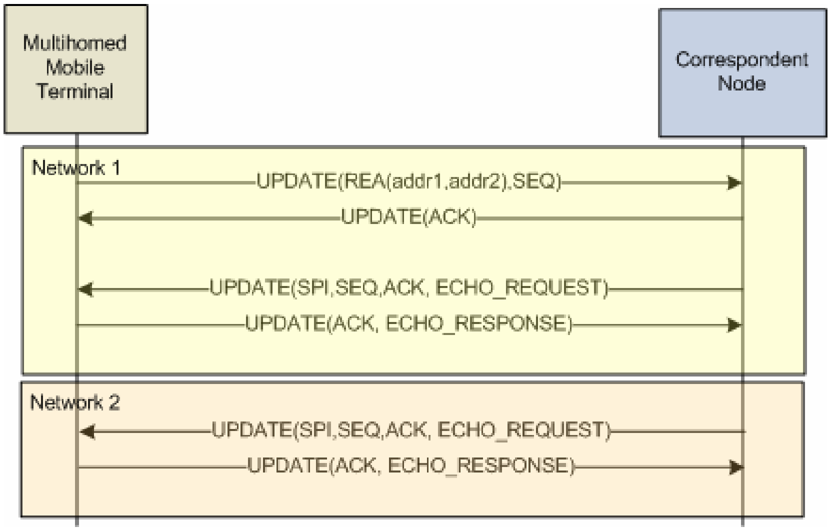

Multihomed Mobile Terminal
Correspondent Node
Network 1
UPDATE(REA(addr1,addr2),SEQ)
UPDATE(ACK)
UPDATE(SPI,SEQ,ACK, ECHO_REQUEST)
UPDATE(ACK, ECHO_RESPONSE)
Network 2
UPDATE(SPI,SEQ,ACK, ECHO_REQUEST)
UPDATE(ACK, ECHO_RESPONSE)

Figure 17 - HIP Multihoming procedure

2.3 QoS and Mobility

The number of mobile users is growing very fast and the higher number of users, the higher the traffic the infrastructures need to support and handle. Whenever there is network congestion, packets are discarded and services sensitive to packet loss and high delays, such as IP telephony and real time streaming, will suffer. Regulating and policing the traffic flowing in the network increases robustness and fairness in its use. The support of Quality of service (QoS) running on top of the base IETF architectures Integrated Services (Intserv) [RFC1633][RFC2212][RFC2215] and Differentiated Services (Diffserv) [RFC2474][RFC2475] aims at being capable of controlling the following parameters: bandwidth, delay, jitter, packet loss and service availability.

According to the mobility protocols described before, when mobile users change their point of attachment, connectivity may be lost until their correspondents are updated with their new location FMIPv6 tries to solve this issue using the make before break paradigm by forwarding the packets from the previous network to the new network, but there is no guarantee that the new network is capable of supporting that amount of traffic. Therefore, none of the described protocols has intrinsic support for QoS.

When QoS is involved in mobility, it should allow users to handover freely between access routers/networks without service degrading and mantaining the original QoS initially assigned.

The QoS infrastructure also performs flow authorization and shapping, handover authorization considering the available resources on the target access router, and it can also be responsible for populating the new access router with the QoS information granted in the previous one. On the other side, QoS infrastructure can also order handovers based on measurements, user policy and service needs. For instance, in terms of measurements, the network detects that the MT is requiring more bandwidth than expected or keeps losing signal. Regarding user policies, an user can have more priority than another on the contracted services, so the network can reallocate resources for another MT in another access router to guarantee that the more priority user gets the requested service with the quality agreed. Finally, in terms of service needs, there should be always room for emergency calls. In this case, some users might need to change its attachment point or be droped in order to assist emergencies.

Proposals to integrate QoS with mobility have been discussed in the literature. [jaehnert05][bless04][marques03] present an end-to-end QoS architecture that enables roaming services over heterogeneous wireless access networks. The proposed schemes are based on a resource manager approach where each autonomous system implements a Domain Resource Manager. The authors present an integrated state model aiming at run time switching between different kinds of handovers in case of failure while preserving reservations.

The integration of mobility with QoS subsystem have been addressed by several EU IST Projects. EVEREST [everest] expects that in future mobile networks several radio access networks may co-exist in the same area and a multi interface capable terminal is connected to the access networks. The overall architecture of Everest is based on an UMTS IP capable core network with QoS support by means of DiffServ mechanisms. As a result, handover decisions are taken by the core network taking into account several parameters, such as, user preferences, terminal capabilities, operator technology selection policies, user profile, location information, networks available and its conditions. AMBIENT networks [ambient] provides an architecture to support heterogeneous technologies, considering user's requirements and mobility aspects with QoS support. The mobility solutions go beyond the traditional handover concept by integrating the session concept to their architecture: it introduces a triggering entity to collect, process and filter triggers that can have an impact on mobility, and a central entity for controlling the movement of terminals across different locations.

There are in the literature multiple references about integration of mobility and QoS mechanisms. Approaches in [marques05] [hillebrand04] [aguiar06] [garcia06] [nursimloo05] [banerjee06] address seamless mobility on heterogeneous operator networks with QoS support, making use of DiffServ architectures which contain a central entity that performs access control, resource reservations and deallocation. All approaches try to overcome the shortcomings of Mobile IPv6 by using mechanisms to predict the terminal movement by triggering handovers and ensuring that the QoS in use is allocated at the new network prior to the terminal's movement. These architectures provide support for intra-domain and inter-domain handovers and mobile initiated handover. [aguiar06] goes a step further in this concept by introducing Network Initiated Handover, where the network is able to instruct the terminal to move to a specific access router if, for example, the access router currently in use is overloaded and some network optimization is need. The mechanism adopted in [nursimloo05] is a virtual spanning tree algorithm based on the terminal's Home Agent where resources are reserved ensuring that wherever the terminal is moving it will have enough resources to keep its current sessions.

Besides these Diffserv-based proposals, there are a large number of approaches integrating mobility and QoS, and some of them also integrating AAA aspects. Some of the proposals only integrate mobility (not always fast mobility is considered) and QoS, through the integration of the mobility signaling with ReSource reserVation Protocol (RSVP) signaling [yasukawa01], [leu03] and [lee04], where RSVP integration with Multi Protocol Label Switching (MPLS) is also a common approach. [leu03] surveys extensively the main problems affecting integration of the RSVP and MIP technology, while [yasukawa01] presents an integrated QoS and mobility scenario. Generally, the main drawback of any RSVP solution is that the reservations need to be allocated along the path between the nodes, which is complex in moving environments.

Due to the amount of increasing multimedia applications for operator networks, there are several studies [nursimloo05][banerjee06][dutta04][banerjee05] that demonstrate the possibility to provide multimedia services (through Session Initiation Protocol (SIP) [RFC3261]) for heterogeneous mobile users while maintaining QoS requirements. All these solutions make use of applications layer mechanisms to minimize the packet loss and service degradation during handover execution. Using a Diffserv architecture for QoS support, [nursimloo05] integrates Fast Mobile IPv6 network layer protocol with SIP to reduce latencies and ensure QoS for the multimedia service. Just before movement, the terminal registers itself at the SIP server in the new domain and issues a re-invite message to the correspondent node through the SIP proxy, which works together with the QoS subsystem. All these approaches make use of a SIP proxy to mediate the session and ensure the QoS needed for that multimedia session.

The use of applications layer mechanisms restricts its use only on session based applications with SIP support. Providing seamless mobility on the network layer enables the use of several application types. The application binds to a single address (MIPv6 case) making the handover completely transparent to the application since QoS assurance and mobility control are performed at lower layers.

2.4 Multicast and Mobility

Internet Protocol (IP) multicast is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to thousands of corporate recipients and homes. Applications that take advantage of multicast include videoconferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

IP Multicast delivers source traffic to multiple receivers without adding any additional burden on the source or the receivers, while using the least network bandwidth of any competing technology. Multicast packets are replicated in the network routers enabled with Protocol Independent Multicast (PIM) [RFC4601] [RFC3973] and other supporting multicast protocols resulting in the most efficient delivery of data to multiple receivers possible. All alternatives require the source to send more than one copy of the data. Some even require the source to send an individual copy to each receiver. If there are thousands of receivers, even low-bandwidth applications benefit from using IP Multicast. High-bandwidth applications, such as Moving Picture Experts Group (MPEG) video, may require a large portion of the available network bandwidth for a single stream. In these applications, the only way to send to more than one receiver simultaneously is by using IP Multicast. Figure 18 demonstrates how data from one source is delivered to several interested recipients using IP multicast.

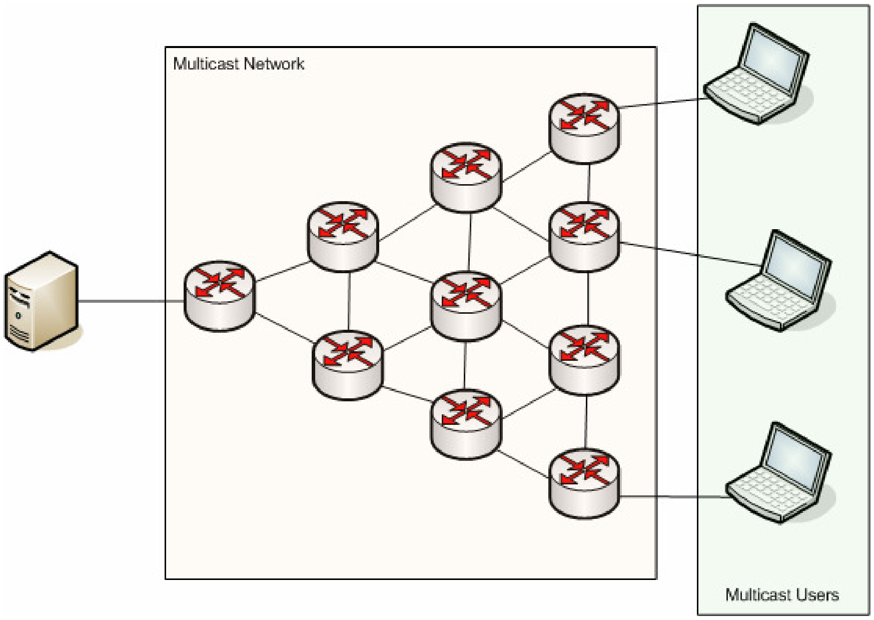


Figure 18 - Multicast Transmission

2.4.1 Multicast Group Concept

Multicast is based on the concept of a group. An arbitrary group of receivers expresses an interest in receiving a particular data stream. This group does not have any physical or geographical boundaries—the hosts can be located anywhere on the Internet. Hosts that are interested in receiving data flowing to a particular group must join the group using

Internet Group Management Protocol (IGMP) [RFC3376]. Hosts must be a member of the group to receive the data stream.

2.4.2 Internet Group Management Protocol

IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Hosts identify group memberships by sending IGMP messages to their local multicast router. Under IGMP, routers listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

2.4.3 Mobility

Regarding mobility as in QoS none of the protocols described support multicast mobility. Due to its multicast nature the destination address in not related with the mobile terminal, which turns out to be very difficult to understand if a specific user is receiving the multicast flow or not. This leads to a major mobility problem, because the user might want to keep the multicast flow after changing networks. In this way, the make before break approach is the best paradigm to use where the multicast flow should already be sent to the new network before the actual handover takes place.

2.5 Summary and Conclusions

Mobile IPv6 is usually regarded as a basic mobility technology for these novel 4G environments, but its inefficiency for seamless handovers is well-known. Fast intra- and inter-technology handovers are a solution to the requirement of seamlessness, and recent works [jaehnert05][marques03] have proposed the integration with QoS control. In order to enhance the Mobile IPv6 operations, the Fast Handovers for Mobile IPv6 protocol was selected. In this protocol a CoA is configured at the MT prior to connecting to the new network, which enables the MT with the new CoA as soon as it connects to the new network. The latency introduced by the BU registration is minimized by the introduction of a tunnel between both new and old access routers.

Mobile IPv6 increased the efficiency of routing between MTs and the correspondent nodes, but binding updates are always sent to home agent and to all correspondent nodes communicating with the MT. To enhance this procedure, Hierarchical Mobile IPv6 use a hierachical distribution of access routers which are able to localize the management of the handoffs reducing the amount of signaling in the wireless network and in the core network, which also increases the efficiency of MIPv6 in terms of handoff speed.

Although Mobile IPv6 is a powerful Internet mobility protocol, it presents some weaknesses for frequently migrating hosts. Specifically, after each host migration, a local temporary address must be obtained and communicated to a possibly distant Home Agent. Cellular IPv6, which is a Mobile IPv6 protocol extension, is again a hierarchical approach combining the efficiency and scalability of IP with inherent features found in cellular networks such as seamless handoff support.

The current Internet uses two global name spaces: domain names and IP addresses, which are used for topological locators and network interface identities. These dual functionalities of IP addresses limit the flexibility of the Internet architecture and are used by transport layers that are bound to IP addresses, which is a problem for mobility and multi-homing. The Host Identity Protocol (HIP) architecture defines a new third name space, the Host Identity name space, that decouples the name and locator roles currently filled by IP addresses, and is used by the transport layer instead of using IP addresses as endpoint names. Regular IPv6 terminals with no particular mobility support are not able to communicate if they do not have support for HIP, which is one of the biggest advantages in the use of Mobile IPv6 based protocols.

Along these lines, previous work [RFC3775][campbell02][RFC4140][RFC4068] has addressed the support of seamless mobility based on the Internet Protocol version 6 (IPv6). While these works have mostly analyzed these mobility aspects conceptually, there is a lack of real evaluations of these mobility platforms.

Preliminary studies [melia04] already demonstrated the feasibility of a platform integrating fast mobility, QoS and AAA. Measurements show that non-mobility aware applications, TCP and UDP based, can provide reliable services, with no packet loss and seamless roaming in this environment.

In the framework of the European IST project Daidalos [daidalos] and of this Thesis, an enhanced IPv6 mobility platform, taking into account QoS and security issues (Authentication, Authorization and Accounting - AAA) was developed. Fast horizontal (i.e. intra technology handover) and vertical handovers (i.e. inter-technology handover) are here doubtlessly beneficial for mobile internet users.

This will be addressed in the following chapter.

Chapter 3 Integrated Fast Mobility

Aiming an integrated architecture with fast mobility mechanisms, the work performed in the framework of this Thesis adopted as its base mobility protocol, Mobile IPv6. Based on this assumption and on the studied protocols on the previous chapter, Fast Handovers for Mobile IPv6 was the inspiration for this work, which extended and adapted this fast mobility mechanism to support QoS and real-time requirements in an operator-driven network. To improve performance two mobility architectures were also proposed, exploiting multicast networks to distribute the traffic among the surrounding access routers.

This work was performed in the scope of the IST-Daidalos [daidalos] project. As such, the architecture addresses both mobile terminal and network initiated handovers in both intra-technology and inter-technology scenarios.

This chapter describes the IST-Daidalos Environment, and the architecture developed for the presented work. Furthermore we describe the modules which compose the mobility architecture regarding: i) handover preparation that discovers handover candidates, inteligently selects the best interface network to use and performs network optimization; and ii) handover execution taking into account the communication with the technology drivers, minimization of packet loss and delays using soft handover mechanisms and the use of context transfer procedures to allow interaction with AAAC mechanisms and provide transparent handovers. The mobility protocol, based on the Fast Handovers for Mobile IPv6 protocol, that was designed and implemented in the framework of this Thesis is then presented followed by the new concepts on using multicast networks to support handovers. Concluding this chapter there is an evaluation of the architectures presented.

3.1 Daidalos Environment

The aim of the Daidalos project [daidalos] is to provide an integrated architecture for multiple access technologies, incorporating wired networks (Ethernet), wireless LANs, broadcast media (DVB-S and DVB-T) and cellular technologies (W-CDMA and TD-CDMA). Daidalos resorts to Mobile IPv6 as a common transport layer, and developments of Daidalos are conceptually technology independent. In addition, Daidalos relies on user-centric concepts, namely, on user profiles. These profiles are associated with specific contractual relations, and adequate privacy management is considered in order to exchange the adequate user information across multiple administrative domains and with content/application providers. The networks support services to customers under a policy based management framework. Figure 19 depicts the overall Daidalos architecture.

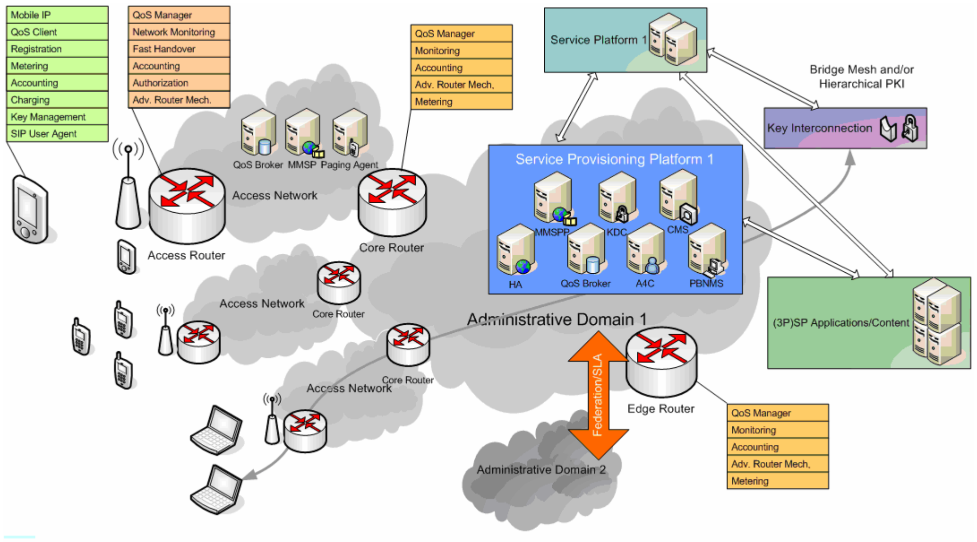


Figure 19 - Overall Daidalos architecture

This section presents the overall network architecture and its mobility sub-system. A simplified vision of the network architecture is depicted in Figure 19 and comprises multiple operator considerations, incorporating access, service provisioning, and content-provider aspects. In the instantiation of the architecture depicted in Figure 19, the incumbent telecom operator still retains most of the functions associated with the network.

The figure mostly represents one administrative domain. This domain may be federated (or have some sort of Service Level Agreements - SLA - established) with other domains (with edge routers in the border). The administrative domain is separated in core and access networks. IP multicast and differentiated services framework are supported across these networks. Three different access networks are represented, all of them with wireless access (for simplicity). The technologies supported are WiFi, Ethernet, WiMax, TD-CDMA and DVB-S/T.

Access networks (AN) are structured in cells. Each cell is controlled by an Access Point - AP (in some technologies this will be a Base Station). Sets of cells are controlled by Access Routers (AR). These routers are the first transport layer control entity, although optimizations at the L2 layer can be done at the AP level.

In each AN, several components may be deployed. Naturally, a manager of the network is in place (the ANQoS-Broker - ANQoSB), for taking admission control decisions and ensure QoS contracts. For wireless environments, and to provide power savings, a paging controller is also present in the network (providing eventually technology independent paging capabilities). For multimedia services a MultiMedia Service Proxy

(MMSP) is deployed in the network, providing the first point of contact for SIP-based services. A CAN (Content Adaptation Node) is also deployed in the AN to perform multimedia codec adaptation if required.

In the core network (CN), interconnected by routers, the operator holds a Service Provisioning Platform (SPP). This platform provides a large set of functions required for efficient telecommunication provision, such as:

- Location server to provide a central repository for querying the operator about a specific user location;
- Global service composer to provide the tools for long-term complex service provision;
- CNQoS-Broker (CNQoSB) to manage the core network transport infrastructure; it will balance resource allocation in the core, according to long-term statistics;
- Home agent to process device mobility;
- MultiMedia Service Provisioning Platform (MMSPP) to support advanced SIP functionalities (registrar, redirect, application servers, etc.);
- Central Monitoring System (CMS) to collect information from probes in multiple entities, providing real time monitoring information;
- Authentication, Authorization, Auditing, Accounting and Charging (A4C) platform;
- Key Distribution Center (KDC) to provide the cryptographic information required for the A4C actions;
- Policy Based Network Management System (PBNMS) to provide the mechanisms for managing the network.

Most of these entities require the cooperation of functions located in different points of the network, or even in the mobile terminal. Many of the functions performed are also highlighted in the terminal, and in the access, edge and core routers (see Figure 19).

This architecture is also able to provide pervasiveness, which is the capability of providing transparent service usage, achieved through the synergistic cooperation of multiple entities. Pervasive components are distributed along all entities, from the mobile terminal to the access router. These are all components that have degrees of personalization and privacy.

3.2 Mobility sub-system

In this section we present a brief introduction to the functionalities of the mobility sub-system. Note that although this architecture was developed for operation with a MIP environment, its development was actually done to provide independency on the mobility protocol to use (so other protocols such as HIP could potentially be used).

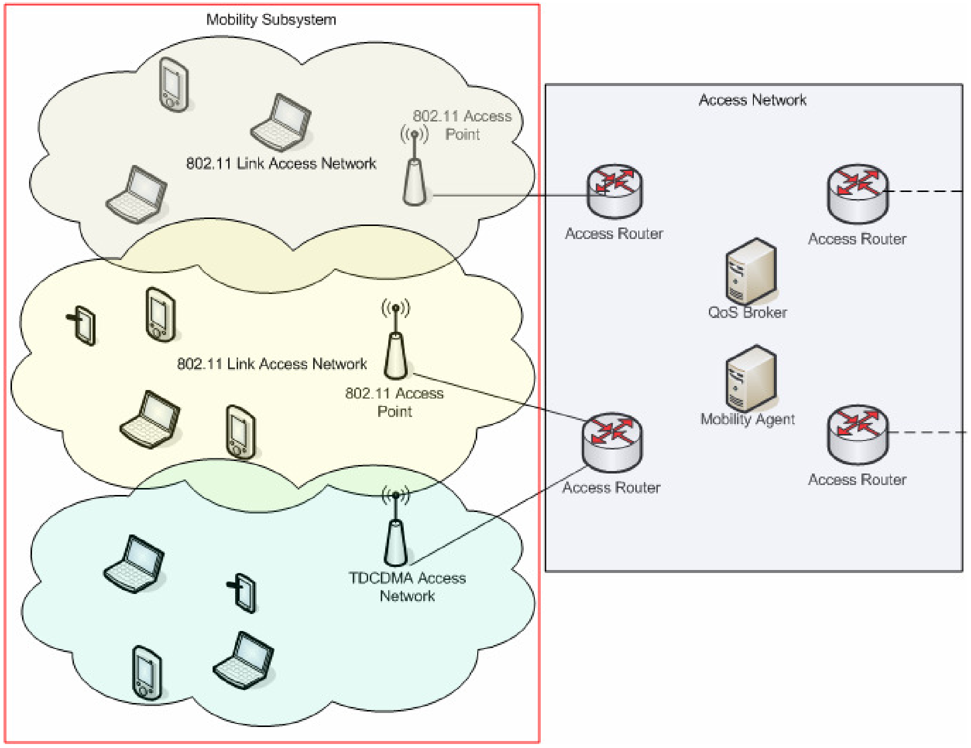


Figure 20 - Mobility sub-system

The mobility [RFC3775] architecture comprises core network and access network components, which implement functions designed to support enhanced fast mobility management, intra- and inter-technology. The mobility management (the sub-system is illustrated in Figure 20) focuses mainly on the preparation and the execution of mobile terminals handovers that can be initiated by the terminal, due to user preferences such as cost, quality and access technology (Mobile Terminal Initiated Handover, from now on referred to as MIHO), or by the network for load balancing and resources optimization reasons (Network Initiated Handover, referred to as NIHO). The NIHO decision takes into account the ”always best connected” paradigm. Main physical components implementing functional entities are the QoSBroker, the access network Access Routers (ARs), access points (APs) and the mobile terminals (MTs). As mentioned, heterogeneous support is provided, through 802.11, TD-CDMA and Ethernet technologies. DVB-T/H is also being addressed in this framework to provide an integrated sub-system, but it is out of scope of this thesis.

There are two phases when an handover occurs: handover preparation where the terminal searches for available networks and requests the handover authorization (if needed); and handover execution where all handover enhancement operations such as duplication and merging of packets and context transfer are being performed, as well as L2 disassociation/association and IP reconfiguration.

The implementation of these modules in the mobile terminal is depicted in Figure 21. Figure 22 presents the mobility entities on the network side, namely on the Access Point, on the Access Router, and on the QoS Broker.

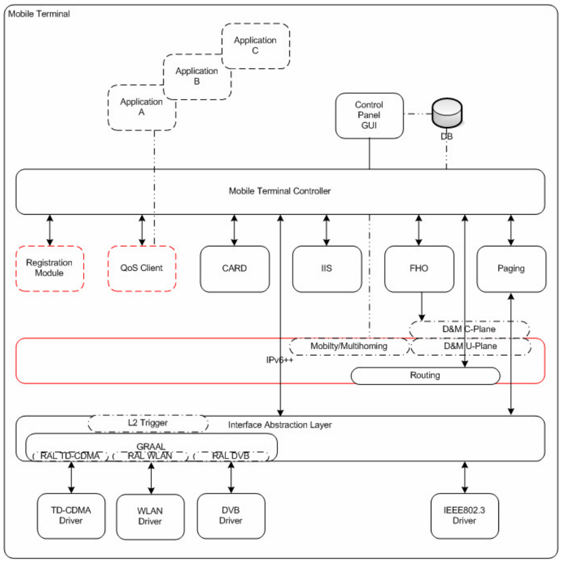


Figure 21 - Mobility sub-system: the mobile terminal

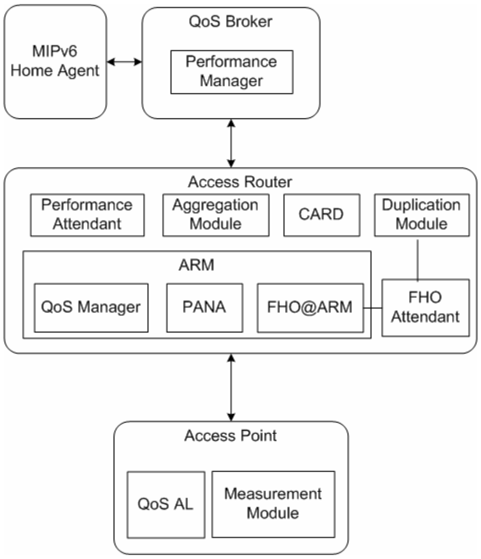


Figure 22 - Mobility sub-system: the network entities

The overall functions of the modules implemented in the framework of the mobility architecture are addressed in the following sub-sections. The detailed description of the modules is then addressed in section 3.3.

3.2.1.1 Performance Management

The modules associated with the Performance Management optimize the access networks' resources in a region controlled by a QoS Broker (QoSB). The handover decisions for resources optimization are taken by both the Performance Manager (PM) module and the QoSB engine module. Available QoS and resources at Access Points as well as data related to signal strength are sent to the PM, which informs the QoSB. This module also accounts for end-to-end QoS information and layer 3 resources to decide on the distribution of the MTs in the access networks and to trigger NIHO for resource optimization.

3.2.1.2 Handover preparation

To enhance the MIHO preparation, parameters beyond the scope of a particular link signal quality need to be available to support an Intelligent Interface Selection (IIS) function on the MT. These parameters are related to the discovery of local (access technology characteristics) and network parameters. This discovery is supported by the Candidate Access Router Discovery (CARD) [RFC4066] protocol. The IIS is the module in the MT that, through the information obtained from the several available access networks, decides on the network the MT will attach to.

The control of the mobility process in the MT is assigned to the Mobile Terminal Controller (MTC), which enables the support of several interfaces in different access technologies through an Interface Abstraction Layer (IAL) function. The IAL implements sub-components to handle wireless and wired access technologies. The control of wireless access technologies is being done with Generic Radio Access Adaptation Layer (GRAAL), which implements Radio Access Layer (RAL) components for each wireless technology, such as TD-CDMA and Wireless LAN (RAL-WLAN). The MTC also interfaces with two external modules: the QoS-Client (QoSC) for handover notifications to the terminal content adaptation function, for resource reservation and for resource re-negotiation when required; and the Registration module for security related control (Secure Neighbour Discovery (SEND) [RFC3971], and Protocol for carrying Authentication for Network Access (PANA) [pana06]).

Although a MIHO is triggered by MTC, several operations are performed previously to the handover trigger, some with the network cooperation. According to periodic reports from the various technologies, IAL aggregates them and passes the information to MTC. With this information and the one on the available access networks provided by CARD, the MTC triggers the IIS to select the interface according to the preferences set by the user and the performance characteristics of the interfaces. In terms of connection preferences, it takes into account the access technology, quality, cost and provider. Also, for the access technology, it is possible to set its order of preference, such as WLAN, TD-CDMA, DVB and Ethernet. Using all these preferences and the data provided by MTC, IIS selects the interface to handover.

In the case of a selected interface different from the current one, or in the case of loss of signal in the current interface (this also triggered the IIS to choose another interface), MTC prepares the handover informing the Registration Module to setup the security in the new link and to get the CoA to use in the new network. It then triggers the Fast Handover (FHO) Protocol.

In the NIHO preparation phase, the PM and QoSB decide on the requirement of a MT handover, based on the procedures previously explained, and notify the MT to move.

3.2.1.3 Handover execution

To perform an efficient handover, while targeting low latency handovers and efficient means to re-establish an MT context on the handover target, a Fast Handover (FHO) operation collaborates together with a Duplicating and Merging (DM) function, as well as with a Context Transfer (CT) function. The FHO operation is based on [RFC4068] with extensions for QoS and security. DM improves performance by duplicating the packets addressed to the MT at the old AR to the new one, to avoid packet loss. CT is used to transfer the mobility related state (including security information) between the old and new AR.

3.2.1.4 Multi-Homing

Within the framework of the mobility process, a Multi-Homing concept, that supports load balancing of the flows between different interfaces for resource optimization purposes, has been developed and realized based on the Mobile IPv6 (MIPv6) protocol and associated network entities. Multi-Homing functional entities are implemented in the MT and with extensions to the MIPv6 HA. The details are out of scope of this thesis.

3.2.1.5 Integration with Authentication and QoS

The Advanced Router Mechanisms (ARM) and the QoSB aid in the support of QoS and Authentication, Authorization, Accounting, Auditing and Charging (A4C) integration. The ARM, located in the AR, provides functionality equivalent to a basic proxy without the need to change any of the legacy applications, and can be considered as a dedicated intelligent transparent proxy [gomes04]. ARM can also perform application to network level QoS mapping for multimedia services, issuing the resource reservation requests to the QoSB and filtering the QoS configurations in the application signalling messages. Finally, it can also receive QoS signalling requests (such as RSVP) and issue the proper resource reservation requests to the QoSB. It also aggregates several other modules such as A4C and Fast Handover Attendant functions which exchange information between them and forward it to their correspondent core network applications.

The QoSB performs admission control and manages network resources; it controls the network routers according to the active sessions and their requirements. As previously referred, it also performs load balancing of users and sessions among the available networks (possibly with different access technologies) by setting off NIHOs. The interaction between QoSB, ARM and mobility modules aim at providing authorized handover process with QoS support.

The architecture also supports reservations in the wireless access part. For this purpose, a QoS Abstraction layer is deployed, which through previous communication with the ARM, reserves resources in the wireless access through QoS Driver modules, specific to each access technology, at the AP and MT.

3.3 Modules Description

Each time an handover occurs, there is a sequence of actions performed by several modules (Figure 21, Figure 22) interacting between themselves in order to provide the best connection available in terms of costs, signal strength and reachability. The handover is divided in two important operations, handover preparation and execution, that determine its success or failure and the capability of being performed without quality degradation and interruption. This section describes the modules associated with those operations.

3.3.1 Handover Preparation

In order to perform a handover some preparation is required. First, we need to discover of the surrounding routers and APs with enough available resources to accept seamlessly the incoming terminal. The target interface needs also to be selected according to the parameters retrieved from the network, by discovery of candidates' ARs, and from the terminal, by measuring the signal strength and identifying the available interfaces. The support for network initiated handovers takes an important role on this architecture; and in order to support them, some mechanisms at the network are required; those include measurements from APs and their aggregation at the AR, which are then collected and conveyed to the Performance Manager, which together with the QoS Broker, will take the decision on which MT to move and where to move.

3.3.1.1 Measurement and Aggregation

The Measurement Module (MM) resides at the Radio Access Point and collects measurements from the radio interface driver. Measurement reports are then sent to the Aggregation Module (AM).

3.3.1.1.1 Measurement Module

The Measurement Module for 802.11x collects technology-specific data received from the measurement interface. Examples of the data include:

- Measurement timestamp
- MT address (MAC)
- Channel
- Signal strength
- Noise level
- Transmission rate (bandwidth)
- Device type (MT/AP)
- Service Set Identifier (SSID)
- AP name
- Beacon interval

3.3.1.1.2 Aggregation Module

The Aggregation Module is comprised of three modules, each having different responsibility.

- APDetector, which is a detector of new Radio Access Points
- APManager, which is a manager of Radio Access Points that are subordinate to a given Access Router;
- AMManager, which is a manager of the Aggregation Module.

3.3.1.2 Candidate Access Router Discovery

The IETF's Candidate Access Router Discovery (CARD) [RFC4066] protocol supports a mobile terminal's handover with two functions. These are: i) reverse address translation of handover candidate APs' Layer-2 identifier (L2-ID) to the IP address of the associated Candidate Access Router (CAR); ii) along with the discovery of individual CARs' capability parameters. The IP address of a CAR is required to initiate a Layer-3 handover, as soon as a Target Access Router (TAR) has been selected out of a list of CARs. The capability parameters, which are associated with each CAR, can support the handover target selection being performed by the mobile terminal's IIS function. For reverse address translation and capability discovery, the mobile terminal utilizes a CARD Request/Reply

(CARD REQ, CARD REPL) protocol message handshake between the mobile terminal and its current AR.

The basic CARD functional entities (FE) are located on the MT and on the infrastructure's ARs to support front-end operation. Information about (Candidate) ARs are cached and maintained in each AR's CAR table. Further FEs might be introduced later to support more enhanced backend operation, like automatic discovery of CARs' L2-ID – IP address mapping information to initialize and maintain CAR table entries associated with the reverse address translation function.

CARD is therefore an essential part of the MT's handover preparation phase in a MIHO scenario.

3.3.1.2.1 Mobile Terminal function

Figure 21 illustrates how the client function of the CARD protocol, which is associated with the mobile device, integrates with the MT architecture. Initiation of the CARD procedure is triggered from the MTC function. The results of the CARD protocol operation are given back to the MTC, which provides consolidated data to the IIS function for further processing. Some access technology specific data is being stored at the MTC for subsequent control of network interfaces through the Interface Abstraction Layer (IAL).

3.3.1.2.2 Access Router function

The CARD function on ARs needs to interface to other local modules to retrieve relevant capability parameters associated with the access network entities. This is required to maintain ARs' CAR tables. An AR needs to maintain its own entries in the CAR table for two reasons: either the AR receives a MT-AR CARD REQ from a connected MT, which requests capability discovery and reverse address translation of an AP's L2-ID associated with the AR, or it receives an AR-AR CARD REQ from one of its neighbouring ARs. In both cases the AR must attach its own capability parameters, IP address information and L2-ID information of associated APs to the CARD REPL message. Figure 23 illustrates the currently specified interfaces associated with CAR table maintenance.

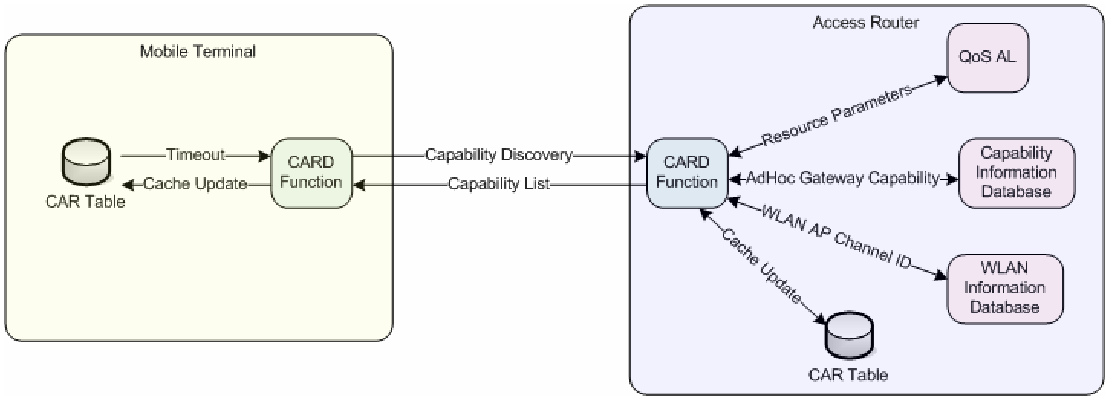


Figure 23 - Illustration of interfaces between ARs' CARD function and local modules where to retrieve the actual capability parameters.

3.3.1.3 Intelligent Interface Selection

The IIS concept aims to organize handover targets according to user preferences and access network environment. The input parameters are signals and measurements performed at the network side, which are provided to the IIS by the MTC, or user priorities captured via a GUI, and cost information.

The components involved in the interface selection operations are the following (

Figure 24):

- The IIS_ALG, set physically at the MT.
- The MTC located at the MT.
- The GUI, located at the MT.
- The IIS_COSTS, set physically at the MT

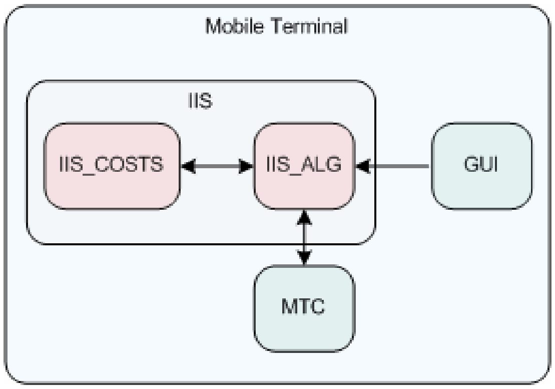


Figure 24 - Intelligent Interface Selection Components

The selection process selects the mobile's local interface (technology), as well as the network's point of attachment, independent of whether it is the same technology as the currently used one (horizontal Handover - HO) or a different one (vertical HO).

There are two different cases where interface selection takes place:

- Application Area 1: Network conditions change. The MTC informs the IIS_ALG about new networks parameters and the selection algorithm is started;
- Application Area 2: User preferences change. The GUI informs the IIS_ALG about user's new priorities and the selection algorithm is started.

3.3.1.3.1 IIS_ALG Module

The IIS_ALG is the module that includes the main intelligence concerning the optimisation process, i.e. the selection optimisation algorithm. The algorithm tries to order the possible handover targets in a list in decreasing priority according to user's preferences. For this purpose, the algorithm needs the relative input parameters and constraints: by using an objective function, it produces the corresponding output. More specifically, the input data is the following:

- The list of available APs with the relative data (provider, link quality, layer 2 address, AR IP address);
- The list of the active interfaces in the terminal (access technology type, layer 2 address);
- The list of the user preferences (the user can order parameters that will thus take a priority for the selection: QoS, costs, provider, technology).

After processing all the input data, the selection algorithm provides as output a list of attachment points in the network (with corresponding network interface in the terminal).

3.3.1.3.2 IIS_COSTS Module

The IIS_COSTS is responsible for estimating costs for different access networks. The cost parameter has a pre-specified value and is used to differentiate the preference of attachment points in the network, instead of another and does not have necessarily price units.

3.3.1.3.3 GUI Module

The GUI is allows the input of the the user preferences list. More specifically, the input data is composed by the following:

- A list of QoS, costs, provider and technology parameters ordered according to the user choice;
- A list of preferred technologies;
- A list of preferred providers.

3.3.1.4 Performance Management

The Performance Management concept aims in keeping the status of the networks and the terminals as optimized as possible, based on measurements and actions performed at the network side.

The components involved in the performance management operations are the following (Figure 25):

- The Performance Manager, located physically at the QoS Broker;
- The Performance Attendant, located at the Access Router (AR);
- The Aggregation Module, located at the AR;
- The Policy Based Decision Enforcement (PBDE), located at the AR;
- The Measurements Module, located at the Access Point (AP).

Beyond these components, there is also a QoS module that is useful for the performance management operation. This module is the QoS Abstraction Layer (QoSAL) located in the AR that communicates with other modules the available resources in the L2 wireless networks.

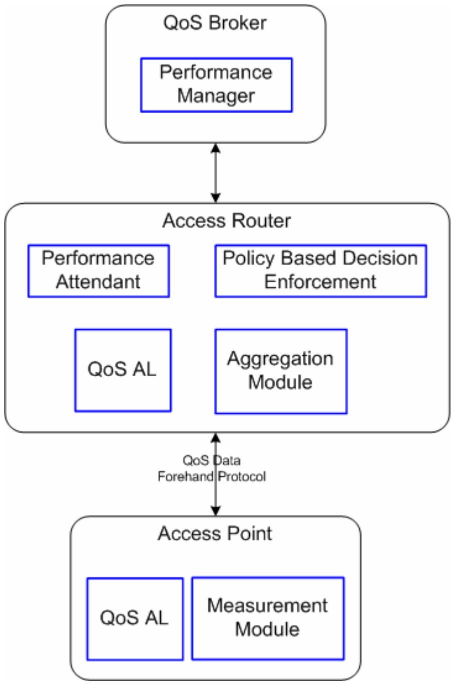


Figure 25 - Performance Management Components

There are three different cases where performance management takes place:

- Application Area 1: Resource optimization for performance reasons. The PA informs the PM for a congestion situation in a specific service area (i.e. some APs).
- Application Area 2: Network initiated handover for mobility reasons. The AM informs the PM (through the PA) for a signal strength degradation concerning specific mobile terminal.
- Application Area 3: Local resource optimization. In this case an intra-AR optimization is performed from the PBDE module.

Network initiated handover for mobility reasons is the second application area within PM. The objective of this application area is to issue a handover initiation trigger from the network. In this scenario, the AM identifies signal strength degradation for a specific terminal. The PM is notified by a specific message, which is passed sequentially through the PA–PM interface. The PM does not reply to this message, but is responsible for triggering the same procedure as in the first application area for this terminal (and not for a set of terminals).

3.3.1.4.1 Performance Attendant module

The PA module is responsible for the following actions:

- Starting the optimization process (application area 1) by receiving the corresponding trigger from the QoSAL module in case of identification of a set of heavily loaded APs. The trigger is passed to the PM module for performing the optimization;

- Starting the network initiated handover procedure (application area 2) by receiving the corresponding trigger from the AM module in case of identification of an individual terminal degradation. The trigger is passed to the PM module;
- Collecting the required parameters from the APs of the AR when requested by the PM module in case of optimization procedure performance.

3.3.1.4.2 Performance Manager module

The PM is the module that includes the main intelligence concerning the optimization process, i.e. the performance management optimization algorithm. As depicted in Figure 25, it is located physically in the QoS Broker, in order to be able to control multiple ARs. The PM holds two interfaces, one with the QoS Broker and one with the PA in the AR, for asking and retrieving all the information needed.

The resource optimization process aims to split the traffic demand among the APs (of the same operator domain) in order to reduce the load in some overloaded APs of the service area. For this purpose, the algorithm needs the relative input parameters and constraints. More specifically, the input data is the following:

- The list of the heavily loaded APs with the relative data (aggregate traffic, available capacity, mean delay);
- The list of the terminals served by the aforementioned APs. For each terminal, the information needed is the terminal profile, the list of visible APs and the list of users. Moreover, each user’s data contains the user profile and the list of services running;
- The list of the cooperating (visible) APs per terminal that are in fact the candidate APs for accepting traffic. Each one is accompanied with the relative data (aggregate traffic, available capacity, mean delay and cost). The cost parameter has a pre-specified value and is used to differentiate the preference of a specific access technology instead of another (although they belong to the same operator) and does not have necessarily price units.

More details about the parameters and the way of obtaining them are given in the sections that describe the interfaces. After processing all the input data, the optimization algorithm provides as output a list of terminals (and also users per terminal) that have to transfer their services (or part of their services) to another AP. This AP may be located under the same or different ARs, but cannot belong to a different domain (as stated before). Moreover, if there is no other possibility, one service may be also forced to use less bandwidth than previously (i.e. to be downgraded). The approval and the execution of the output is a QoSB’s responsibility.

3.3.2 Handover Execution

The handover execution is the critical operational phase of this architecture. It has to be performed as fast as possible in order to avoid service interruption or noticeable degradation on the ongoing flows. Since it is time critical the modules developed to support handover execution need to be synchronized and quickly react to events. The handover execution phase is controlled by the Mobile Terminal Controller (MTC), which acts as a coordinator of the mobility between access technologies and the Fast Handover Module (FHO). Interface Abstraction Layer (IAL) abstracts the current interfaces and technologies to upper layers by receiving the aggregated information of the interfaces from Generic Radio Access Adaptation Layer (GRAAL) and issuing commands requested by MTC to the drivers. That information is collected from the different Radio Access Layers (RAL) and enforced at the driver, one for each technology. During the handover period a function named Duplication and Merging (DM) takes care of the duplication of packets in the old link to the new link, while the packets are merged at the terminal to remove the duplicated packets. A Context Transfer (CT) function is responsible for setting up the user context from the old AR to the new AR.

3.3.2.1 Mobile Terminal Controller (MTC) function

The MTC is the high level contact point of several modules, the IAL, the CARD, the IIS, the FHO, the RM and the QoSC, and resides in the MT. It provides control for these modules and supports their functionality by coordinating the information flow between them. Figure 21 depicts the MTC and its interfaces to other modules which also reside in the MT.

The MTC is able to hold and manage the following information:

- Parameters describing the MT’s current status: the L2 ID of the interface in use, the L2 ID of the AP in use, the IP address of the AR in use, the IP address of the MT over the interface in use (CoA), and a list of the services that are running on the terminal and the corresponding QoS levels being provided to each of these services;
- Parameters describing the handover target’s characteristics: the L2 ID of the new interface, the L2 ID of the new AP, the IP address of the new AR and the MT’s new CoA;
- A list of available APs that the MT perceives, each entry comprising: the L2 ID of the AP, the L2 ID of the corresponding interface, the technology type of the corresponding interface, the signal quality perceived from the AP and the ID of the corresponding network provider;
- A list of candidate ARs, each entry comprising: the IP address of the AR, its certificate and a list of its associated APs and their characteristics (the L2 ID, the number of available channels and the available bandwidth of each associated AP);
- An ordered (by order of preference) list of candidate ARs, each entry comprising: the IP address of the AR, the L2 ID of a preferred AP and the L2 ID of the corresponding interface.

The MTC is responsible for ordering the selection and de-selection of the appropriate attachment point via its interface with the IAL at power-up and at handover execution. It is also responsible for triggering CARD and IIS operation upon indication that the perceived signal quality is degrading (mobile terminal initiated handover scenario) and for handling handover messaging with the FHO module residing in the MT, both in the cases of MIHO and NIHO. Lastly, it interacts with the RM and QoSC modules.

Figure 26 illustrates the integrated message sequence chart involving modules residing in the MT, in the case of a MIHO. This sequence chart contains all the processes and messages needed to perform an MIHO, starting by a IAL_MEAS_REPORT which indicates the presence of a new AP or a signal degradation of the current AP. MTC upon receiving this message contacts CARD in order to find information related to the surrounding ARs and possible candidates. The output of CARD operation is a list of candidates that will be fed into MTC which will require an measurement of each one by issuing an IAL_MEAS_REQUEST. After the response, IIS is triggered in order to select a list of up to three candidates ARs and APs which will serve as input to the FHO client.

The FHO client module is then triggered and will ask the network for handover authorization. Once the handover has been authorized, reservations on the new access router performed and duplication of packets made, the MTC will start merging the duplicated packets and will ask the RM to provide with the new CoA which is a cryptographic generated IPv6 address (CGA) calculated according to the information provided by A4C subsystem.

The FHO_HO_REPORT message will triggger the FBU message sent by the FHO client module in order to inform the network about the movement of the terminal. After sending this message, FHO will reply back with the indication that the L2 handover can be performed by sending a FHO_L2_DISC to the MTC. MTC will then forward this information to the IAL which will activate the new interface or perform the change of AP together with the RALs of each technology. As soon as the L2 connection is performed IAL will send the status of the attachment to the MTC that will inform FHO client that the connection to the new AP is performed thus the information about our presence on the new AP can be sent, by means of a FNA message. Concluding this operation, a FHO_STOP_MERGING message is received by the FHO client module and forward to the MTC indicating that the handover is completed, the reservations on the old access router are freed and the duplication of packets are no longer being performed. The merging of packets is also stopped and IAL is informed of the success of this operation which will then turn off the previous interface if this was an inter-technology handover. Finally, RM module is also informed about this process and can resume its SEND operation.

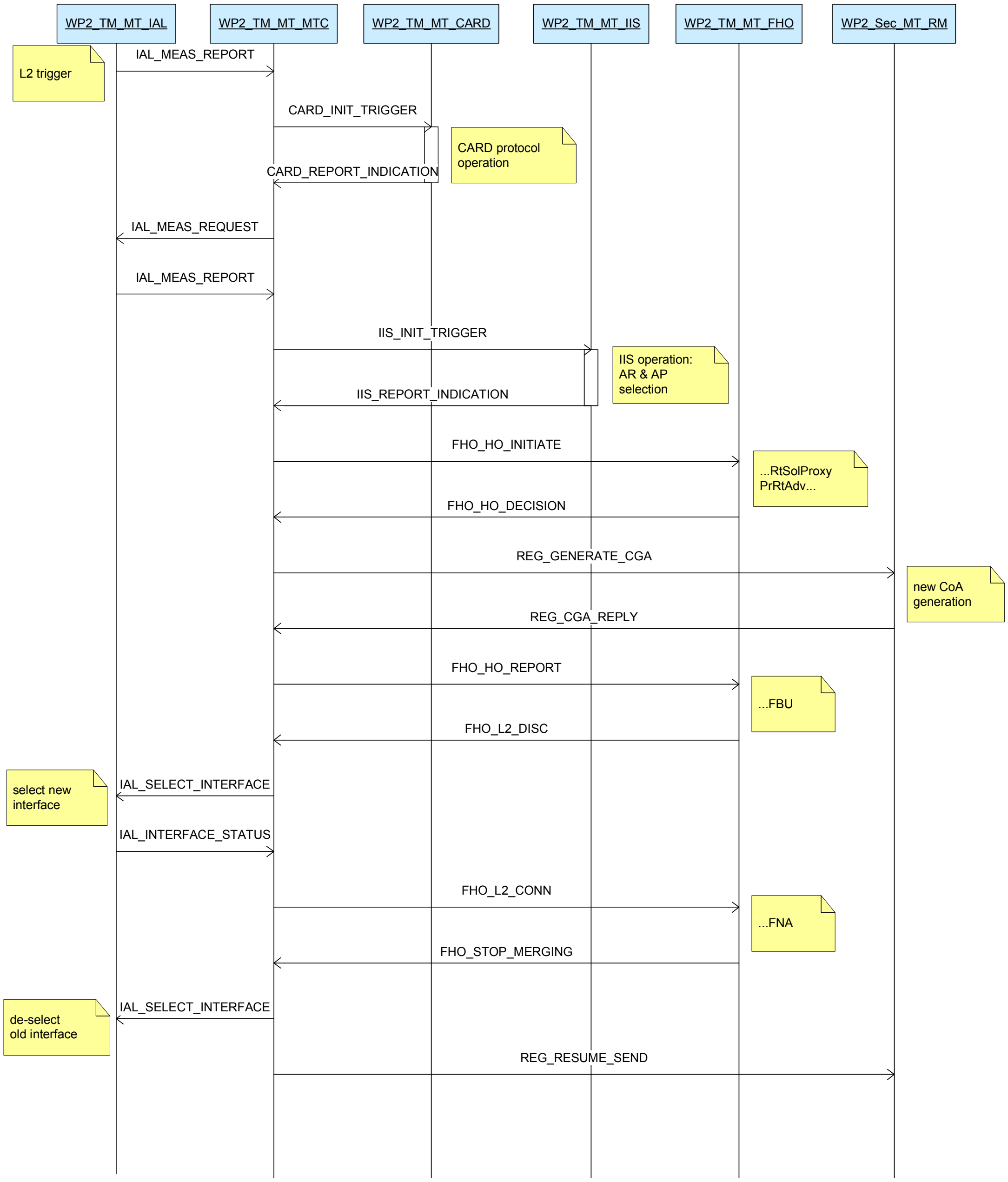


Figure 26 - Integrated message sequence chart (MIHO)

3.3.2.2 Interface Abstraction Layer Module

The main role of the Interface Abstraction Layer, denoted as IAL, is to hide the nature of the access technologies to upper layers. It provides the MTC with the information about layer 2 identifiers and measurements about signal quality of the access points.

The IAL controls the resource managers of the different technologies (WLAN, TD-CDMA, DVB-T, Ethernet, etc.). As shown in Figure 21, the IAL interacts with the access technologies using some specific intermediate modules. The Radio-based access technologies are controlled through a Generic Radio Access Adaptation Layer (GRAAL) module, while wired technologies are directly controlled by the IAL.

The IAL is able to detect and identify interfaces and provide a unified view of them to upper layers in a generic manner, in order to hide specific characteristics of access technologies. It then provides triggers for interface up/down events which can be used as synchronisation triggers for starting interface selection and/or network registration.

The main functionalities of this module are:

- Identification and assignment of a device identifier to an interface as it becomes available. The device identifier qualifies the interface's technology and product identity and can be used to determine appropriate technology specific actions when necessary;
- Provision of triggers when changes in the status of an interface occur, e.g., when the interface goes up (e.g., PCMCIA card insert, iwconfig wlan0 essid ap-daidalos1) or down (e. g. PCMCIA card remove, 'ifconfig eth0 down');
- Configuration and delivery of reports of available channels on an interface along with their connection strength and quality measurement and frequency;
- Getting and setting generic and or specific device configuration parameters.

The measurement functionality contains two main functions:

- Configuration of the measurement reporting to be performed by the MT to the network;
- Generation and delivery of measurement reports by the MT to the network.

The delivery of measurement reports from the IAL to the MTC could be on-demand, or event-based. The event could be a signal which is fading or completely lost.

At the power-on, the IAL sends an unsolicited measurement report to the MTC that contains the list of APs perceived by the MT from the available drivers and their signal quality.

3.3.2.3 Generic Radio Access Adaptation Layer Module

The GRAAL module manages interfaces that use radio-based access technologies like WLAN, TD-CDMA, and DVB-T.

The GRAAL can receive a list of APs to scan. Otherwise, when requested to make measurements (with a list of network provider IDs), the GRAAL replies with all the parameters relative to all the interfaces, whether at power-on or handover.

This module gets technology-specific measurements, such as SNR, from RAL components and translates them to a generic format. It also determines the quality of the mobile to network signal strength based on a pre-configured and updatable table per radio access technology. Furthermore, it retransmits each request received from the IAL module to the corresponding radio-based RAL component.

The GRAAL module interacts with three independent modules that interface with drivers:

- RAL-WLAN: this radio access layer controls the WLAN driver;
- RAL-TD-CDMA: this radio access layer controls the TD-CDMA driver;
- RAL-DVB-T: this radio access layer controls the DVB-T driver.

When the GRAAL module receives a measurement request from the IAL containing the list of APs to be scanned it sends a measurement request to corresponding RALs. When it gets the measurement reports from all requested RALs, it merges and maps them to a uniform format and sends back to the IAL a complete measurement report.

3.3.2.3.1 RAL-TD-CDMA Module

The RAL-TD-CDMA is the module that manages the radio operation of the TD-CDMA medium. It provides measurements requested by the GRAAL module after collecting them from the TD-CDMA driver. It also activates and de-activates the TD-CDMA interface according to requests coming from the GRAAL module.

3.3.2.3.2 DVB-T Module

The RAL-DVB-T is the module that manages the radio operation of the DVB-T medium. It provides measurements requested by the GRAAL module after collecting them from the DVB-T driver. It also activates and de-activates the DVB-T interface according to requests coming from the GRAAL module.

3.3.2.3.3 RAL-WLAN Module

The RAL-WLAN is the module that manages the radio operation of the WLAN medium. It provides measurements requested by the GRAAL module after collecting them from the WLAN driver. It also activates and de-activates the WLAN interface according to requests coming from the GRAAL module.

3.3.2.3.4 WLAN-802.11b-driver

This module is the driver that controls the WLAN 802.11b card. It is based in the hostap driver (http://hostap.epitest.fi/).

This module provides the following functionality to a control program (the RAL-WLAN module):

- Setting up a mask of channels to be scanned;
- Scanning of a number of channels defined by the channel mask. The result is the APs found (identified by their MAC address) and the signal level in the communication with each one;
- Execution of a L2 handover to a certain AP (defined by its MAC address).

3.3.2.4 Duplication and Merging : D&M Modules

The D&M function operates during the FHO execution. It is responsible to duplicate the IP flow in the old AR and to merge it in the MT. It allows duplication and merging of IP flows during handover without the need to synchronise duplicated-flows transmission for the downlink traffic. According to the Daidalos architecture, it has been estimated useless to implement this function for the uplink traffic. This function is based on three main processes: the registration process, the duplication process and the merging process. The main idea is to setup two tunnels from the old AR to the MT, one from the old point of attachment and the other one through the new Access Router.

3.3.2.4.1 Mobile registration process

In order to be connected to several ARs, the MT must be logically associated with as many Care of Addresses (CoA) as ARs, since each CoA identifies a link to the MT through a unique AR.

3.3.2.4.2 Duplication Process

In order to duplicate packets, the D&M intercepts packets sent by the Correspondent Node via some Netfilter hook, extracts from each packet the destination address and finds the corresponding new CoA. Using those CoAs, the D&M agent creates new IPv6 packets with the same payload information, but with substitute CoA as new destination address. A sequence number field is inserted in each IPv6 packets header as a specific extension header, the DIO (Duplication Information Object). This sequence number is used to number all packets sent through this tunnel. The duplicated packets will be identified by the same sender, same receiver and same sequence number fields. Duplicated and numbered packets are then tunnelled to the MT via corresponding ARs.

3.3.2.4.3 Merging process

The use of the D&M process to send separate copies of the same data via multiple ARs to the MT, introduces the need to filter the duplicated packets. To perform this filtering, the D&M function needs to match those multiple streams in IP layer at reception. It intercepts all tunnelled packets with the DIO extension header and checks if the sequence number is included in the IP packet. If there is no sequence number, which means that this IP packet was not duplicated, the process will route normally the payload information. If the sequence number is included in the packet and the source-address has an entry in D&M table, the packet has been duplicated. If the sequence number is listed as a received packet in table, the IP packet will be discarded (it has already been received).

The D&M function is triggered by the IAL in the Mobile Terminal, when the handover is ocurring. It directly interfaces the FHO function in the AR.

3.3.2.5 Context Transfer Function

The transfer of session context may be advantageous in minimizing the impact of mobility on, for instance, QoS, A4C/security, or robust header compression state. Context transfer can be used to replicate the configuration information needed to re-establish the respective protocols and services. In addition, it may also provide the capability to replicate state information, allowing stateful protocols and services at the new node to be activated along the new path with less delay and less signalling overhead.

This function is integrated in the overall subsystem in order to support and enhance intra- and inter-technology handovers. The function is designed in a generic fashion, which makes it usable as a transfer tool between any two entities, provided the necessary integration is realised through the defined interfaces. Here we present the generic framework and its application [vieth].

The following entities are defined for the Context Transfer function:

- Originating CT user: this entity holds the context information and can trigger CT;
- CT sender: serializes context from C++ Objects to protocol objects and communicates with CT receiver using the Context Transfer Protocol, defined below;
- CT receiver: deserializes and delivers context to the target CT user;
- Target CT user: the entity receiving context. It is up to this entity to activate (or not) the context at the appropriate moment. This entity can also request context through a request trigger;
- Trigger entity: responsible for triggering context transfer (this can be the Originating CT user or an external entity, such as the FHO module).
- Activation entity: responsible to inform CT receiver that contexts can be delivered to Target CT User.

The messages defined by the CT protocol are:

- Trigger CT – Triggered by an entity which understands the need to transfer information about a particular terminal to the new location;
- getContext – The CT sender module asks for the context to be transferred to the originating CT User;
- Start CT – The context asked by the previous message is delivered and CT can start to be transferred;
- CT Data – The data being tranferred is sent from the CT Sender to the CT Receiver;
- CT Request – CT Receiver may request the context to be transferred (p.e. reactive handovers);
- CT Data Reply – Aknowledges the reception of the CT Data message;
- Activate Context – The context activation is triggered by the Activation Entity which has the knowledge of the handover;
- Received Context – The received context is then delivered to the Target CT User which will receive and activate this context.
- Request CT – The Target CT User can also, optionally, request for specific context by issuing this message.

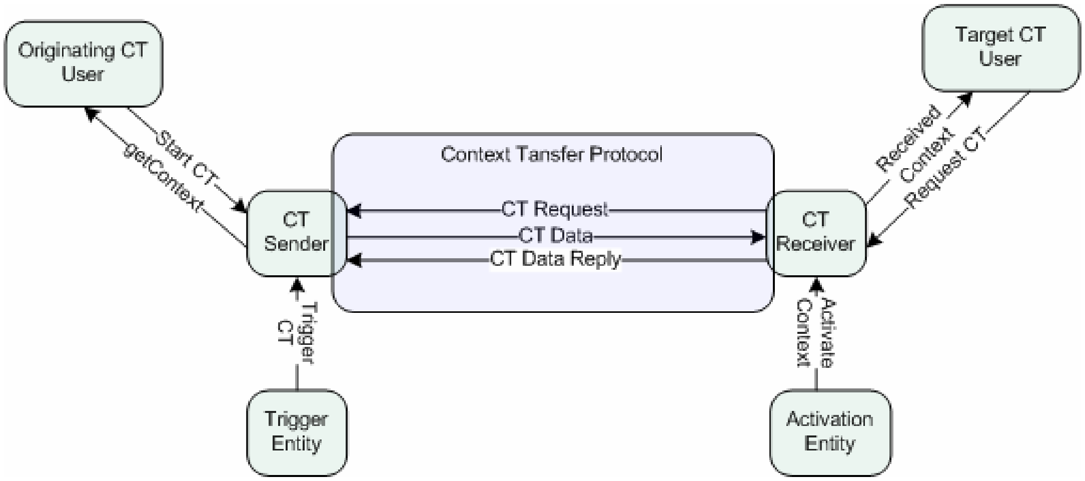


Figure 27 - Context Transfer entities.

3.3.2.5.1 Integration with Fast Handover

The CT function is instantiated in ARs and integrated with FHO module. The FHO module in pAR acts as a Trigger entity, while the FHO module in nAR acts as an activation entity. For FHO integration, one of the contexts included in this message is a HO Session-id, which is transferred to the nAR (CT receiver). When FHO module in nAR receives the first COPS message (which installs queues) from the QoSB, it obtains the HO session-id, which is delivered to the CT module.

When the contexts, regarding A4C and multicast information, are received at the nAR, the CT module checks if it can deliver the contexts for activation (that is, if the HO session-id is present). As a fallback, when the nAR receives the FNA, it sends an activation trigger to the CT module at the nAR, containing the handover session-id.

3.4 The Mobility Protocol

This section describes the proposed Fast Handover Protocol and its functionality and operation both on the MIHO and NIHO modes.

3.4.1 Fast Handover (FHO) function

FHO for Mobile IPv6 is an IETF RFC experimental standard providing seamless mobility and low probability packet loss to devices roaming across heterogeneous access networks. In the scope of this Master Thesis this protocol has been extended and enhanced to better meet QoS requirements. Therefore a two level architecture (i.e. ARs and QoSB) has been designed and specified. The ARs in the current and new networks are responsible for the mobility mechanisms to provide fast mobility. The QoSB entity has been integrated in the fast mobility process to allow for admission control, authorization, network resources optimization, and resource reservation in the new network to provide seamless mobility. The protocol further provides a flexible way to manage both MIHO and NIHO.

Herein, the module implemented by each MT is called FHO client, whereas the module located in the ARs is denoted as FHO attendant. The communication over the wireless medium is done via ICMPv6 messages and the communication with the QoSB is performed via Common Open Policy Service (COPS) [RFC2748] messages. The different messages are classified as follows:

RouterSolicitationProxy (RtSolPr) ICMPv6 - This message initiates a MIHO. The datagram is sent to the current AR (oAR) containing a list, ordered by preference, of up to 3 possible candidate ARs. This message has a flag indicating if the handover is imminent. The handover might be initiated because of mobility reasons or as a consequence of user preferences.

ProxyRouterAdvertisement (PrRtAdv) ICMPv6 - This message contains the selected candidate AR. This indication could be solicited (e.g. reply to an RtSolPr in case of MIHO) or unsolicited in case the network initiates a handover (NIHO).

FastBindingUpdate (FBU) ICMPv6 – The MT can stop the handover process (by a specific flag), or indicate that it is going to change its point of attachment.

FastNeighborAdvertisement (FNA) ICMPv6 - After changing its physical connection the MT has to advertise its presence on the new link in order to populate the nAR's IPv6 neighbour cache. This step is important for packet delivery.

HandoverRequest COPS - In order to request a handover admission the oAR sends the list of candidate ARs to the QoSB for validation. Tipically, the first entry in the list matching network resource allocation is then selected as the new AR.

HandoverDecison COPS - QoSB advertises the handover candidate AR to the old AR.

HandoverReport COPS - This message is used for reporting purposes.

CopsDecision COPS - By means of this message, the new AR is informed about the new MT roaming. Policies rules are therefore installed and the context for the specific MT is activated.

CopsReport COPS - The nAR inform the QoSB about a successful handover.

3.4.1.1 Mobility Process

The mobility architecture provides the flexibility for handover decisions triggered either by the terminal (MIHO) or the network (NIHO). In this section we describe the overall process for triggering the two types of handovers (NIHO and MIHO) and the associated mobility execution.

3.4.1.2 Mobile Terminal Initiated Handover Operation

This section describes the mobility process for MIHO operation. This process is depicted in Figure 28.

Although a MIHO is triggered by MTC, several operations are performed previously to the handover trigger, some with the network cooperation. According to periodic measurement reports from the various technologies, IAL aggregates them and passes the information to the MTC. With this information and the one on the available access networks provided by CARD, the MTC triggers the IIS to select the interface according to the preferences set by the user and the performance characteristics of the interfaces. In terms of connection preferences, it takes into account the access technology, quality, cost and provider. Also, for the access technology, it is possible to set its order of preference, such as WLAN, TD-CDMA, DVB and Ethernet. Using all these preferences and the data provided by MTC, IIS selects the interface to perform handover.

In the case of a selected interface different from the current one, or in the case of loss of signal in the current interface (this also triggered the IIS to choose another interface), MTC prepares the handover informing the Registration Module to setup the security in the new link and to get the Care of Address (CoA) to use in the new network.

After this process the MTC triggers the Fast Handover (FHO) Protocol. The handover operation may not proceed without previous authorization {*Mobile Terminal Handover Authorization*} from the network in order to guarantee that resources and authorizations are available at the new attachment point. Therefore, FHO protocol conveys the handover request to the ANQoSB (QoS Broker in the figure) by sending ICMPv6 messages to the FHO Attendant at the current AR (*Router Solicitation for Proxy* and *Proxy for Router Advertisement* messages). This handover request is then forwarded to the ANQoSB in terms of COPS Messages (*Handover Request* and *Decision* messages). After checking resources and authorizations to connect to the requested AP/AR, the ANQoSB issues the decision to the new AR to make it aware of this handover {*Access Router Handover Authorization*}.

The new AR can then set QoS reservations to the flow(s). At this point in time, all traffic directed to the current CoA is duplicated, by the Duplication and Merging function, at the current AR to the new CoA minimizing and/or avoiding any packet loss. The MT, upon receiving the *Proxy Router Advertisement* message starts the merging process to avoid the existence of duplicate packets. Upon the reception of the handover decision at the terminal, if handover is allowed, the FHO client instructs IAL to check if the AP selected by the ANQoSB is still available {*Mobile Terminal L2 Availability*}. The FHO Client will then inform the network about Link availability and then will instruct IAL to change to the new AP if the link is still available (*Mobile Terminal L2 Handover*). After this change {*Mobile Terminal Handover Execution*}, the terminal informs the new AR of its presence through a *Fast Network Advertisement* message, finishing in this way the handover procedure {*New Access Router Complete Handover Time*}. After handover has completed successfully, the ANQoSB informs the old AR through an *Handover Status Decision* message {*Old Access Router Handover Execution*} to delete the QoS reservations and stop the duplication of packets. At this point the old AR sends a message to the MT instructing it to stop the merging process. {*Old Access Router and Mobile Terminal Complete Handover Time*}.

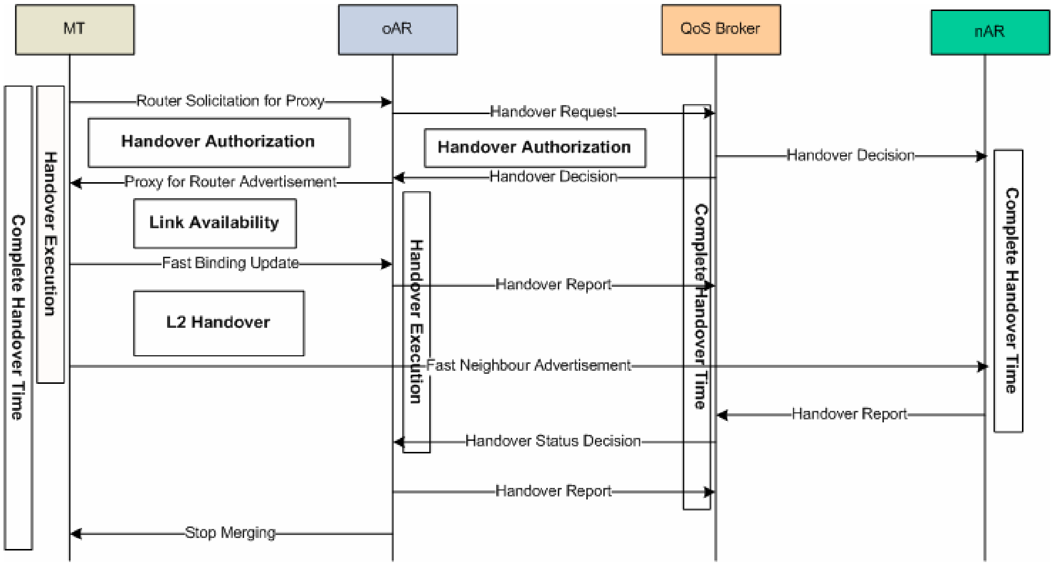


Figure 28 - Fast Handover Signalling

3.4.1.3 Network Initiated Handover Operation

In many situations, MTs do not have all the information required for an optimum handover decision. For instance, APs have more transmission power than terminals, causing loss of connectivity before the terminal realizes it. By monitoring the signal strength reaching the APs (from the MTs), it is possible for the AP to predict a loss of connectivity before it occurs. Another example is an excess of terminals attached to an AP that can cause degradation on the link quality for all; or a MT starting a demanding application that may leave no bandwidth for others. These examples show why NIHO support is essential and should be enforced by the network.

The network was built with these potential issues in mind: the Performance Manager module, with the information collected from the different APs and ARs, runs an algorithm to decide the best distribution of the MTs in the network to optimize network resources. After selecting the action to be taken, typically deciding which MTs to redirect to a new point of attachment, the Performance Manager sends this information to the ANQoSB, which then triggers the handover process. The handover execution then follows the same procedure as in MIHO, starting with the *Handover Decision* message from the ANQoSB.

3.4.1.4 Handover between Different Access Networks

Notice that, although not shown in Figure 28, in the case of the old and new ARs belonging into different access networks, when the ANQoSB in the old network receives the *Handover Request* message, it needs to contact the ANQoSB in the new network to ask for available resources and to transfer the context related to users, sessions and QoS (as proposed already in [aguiar06] and [marques03]). The new ANQoSB answers with the resources availability and the old ANQoSB can then send the *Handover Decision* message to the old AR. This situation is shown in a simplified manner in Figure 29.

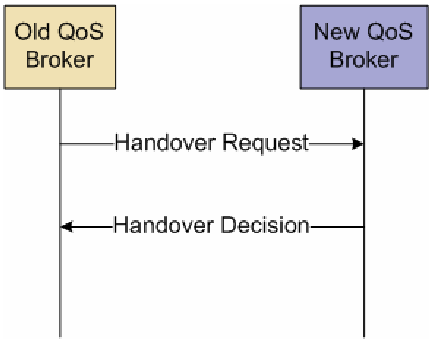


Figure 29 - Handover inter-QoSB areas

The process, in the case of inter-domain handover, is even more complex, and instead of the inter-QoSB communication, it requires interconnection of the Authentication entities (A4C). In both inter-QoSB and inter-domain handover, specific protocols are used between the entities, in order to optimize the handover processing.

3.5 Handover Supported by Multicast Networks

Multicast is used to deliver the same information to different destinations without wasting network resources. This characteristic of replicating the same information can be used to enhance the mobility process, as the information can be sent in advance to both the previous and new ARs through multicast.

In this section we study the possibility to extend the previous proposed architecture to make use of multicast networks to support fast mobility, reducing latency and packet loss. It can also provide more flexibility to the terminal in terms of selecting the appropriate candidates, since these only need to be chosen in advance to belong to the same multicast group. This section presents two proposals: the first one is integrated with QoS and is an extension to the previous architecture; the second provides simple fast mobility without any QoS service support to provide low latency and packet loss in high mobility environments.

3.5.1 Fast mobility supported by a Multicast Network with QoS integration

Multicast networks are the best choice to transport the same traffic inside a network without using duplication mechanisms. With the assumption of a multicast network and the previous fast mobility process in mind, an integrated architecture was designed. The integration of these two techniques includes an extra step in the target selection. The MT does not need to rely on additional protocols to discover surrounding networks, which is a time consuming operation and may cause an interruption on the current connectivity (since it has to disconnect, survey the wireless channels and connect again). This operation is done by the network: since we are considering handovers inside the same domain, the network administrator has the complete knowledge of the network topology. Using this knowledge, the administrator can configure the QoSB with the network topology. The QoSB can then select the proper surrounding AR when a HandoverRequest is made, both by MIHO and NIHO approaches.

In order to guarantee that no packet is lost in this process, several source specific multicast networks are established using the known network topology; this source specific multicast network is formed by each AR (the source) and its surrounding neighbours at the network's boot up.

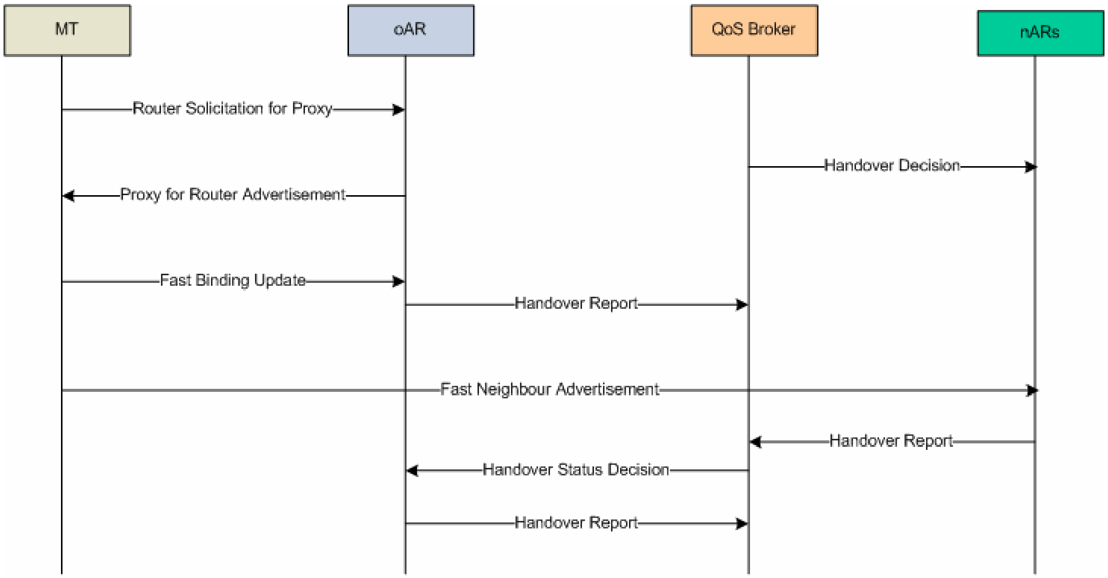


Figure 30 - Fast Mobility Scheme supported by a Multicast network integrated with a QoS System

This fast mobility process is depicted in Figure 30. The MT sensing lower signal in the current AR, performs a handover request {*Routing Solicitation for Proxy*} directed to the AR which is then forwarded to the QoSB {*Handover Request*}. The QoSB looks up the MT surrounding networks, checks which networks can handle the current MT's connections and answers back with the possible targets {*Handover Decision*}. These targets are then informed of the possible handover, and that they are candidate targets and need to be prepared to handle the MT {*Handover Decision*}.

The current AR, upon the reception of this response from the QoSB, forwards the information to the MT {*Proxy for Router Advertisement*}. If the handover is allowed, it starts to intercept the traffic directed to the MT inserting it in the previously established source specific multicast network. At this point, all the surrounding ARs receive the traffic directed to the MT, buffering it for delivery when the MT attaches or until they are instructed by the QoSB that the handover procedure is complete. At this point, the MT can now freely move to any of the candidates listed by the QoSB. Before leaving its current network, the MT sends a *Fast Binding Update* to the AR which is then reported back to the QoSB {*Handover Report*}. As soon as it is attached to a network it sends a *Fast Neighbour Advertisement* and the AR starts delivering the packets to the MT which also needs to send a Binding Update to its correspondent nodes (not depicted for readability issues). The correspondent nodes then send the Binding Update Acknowledgment to the MT. At this moment the AR triggers the information of reception of the MT to the QoSB {*Handover Report*}. The QoSB forwards this message to the previous AR {*Handover Status Decision*} in order to inform if the handover was successful and, if so, to stop inserting any remaining traffic in the source specific multicast network. The previous AR reports the successful handover back to the QoSB {*Handover Report*}. Each of the ARs informed of the handover have a handover time frame for its success; if the FNA message does not arrive in that time frame, the buffered packets are discarded and they start discarding all the incoming traffic directed to the MT.

After all these steps, the MT is directly communicating with its correspondent nodes with no interruption of the current communication.

In a NIHO scenario, the MT receives the order to move to one of the candidate targets following the previously described procedure.

3.5.2 Simple Fast mobility supported by a Multicast Network

The fast mobility mechanism presented in this section is targeting to a fast mobility network, where MTs are always moving with a very high probability to be in a low signal coverage or in overlapping areas. With these requirements, a fast mobility scenario without any intervention of bandwidth management mechanisms was designed. This mechanism is presented in Figure 31.

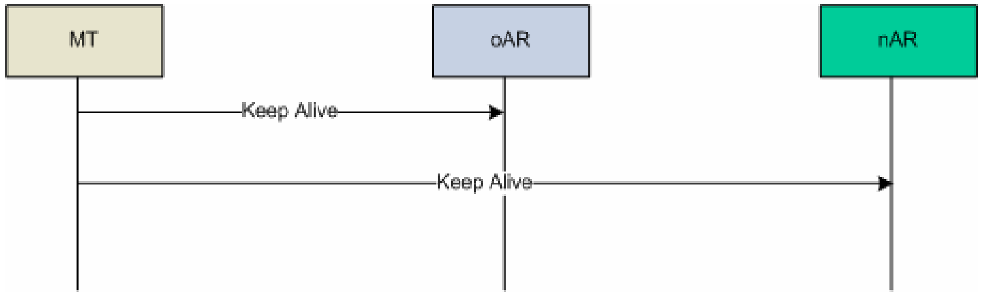


Figure 31 - Fast Mobility Scheme supported by a Multicast network

As in the previous presented solution, there is a source specific multicast network previously established between each AR and its neighbours. All the traffic directed to a MT is intercepted and inserted in the multicast network corresponding to that AR. At this point, all its neighbours receive the traffic, which has a small lifetime in the surrounding ARs buffers. For a small period of time, the ARs keep the traffic in order to guarantee the delivery to the roaming MT as soon as it attaches and signals the attachment. Periodically, the MT sends *Keep Alive* messages to its current AR. These messages signal the AR that the MT is still connected to that AR; as long as the AR is receiving this signal, it inserts the traffic into the multicast network. When the MT moves, it senses new ARs. To attach to a new AR, the MT signals the attachment with a *Keep-Alive* message. It also sends the BindingUpdate message to the CNs. When an AR receives the Keep-Alive message, it starts to insert the unicast traffic directed to the MT into its source specific multicast group, preparing a future handover of the MT. After a time out of 3 Keep-Alive messages, the previous AR stops introducing the traffic into the multicast network. At this stage, the handover procedure is concluded.

3.5.3 Architectures Evaluation

The three fast handover mechanisms presented (3.4, 3.5.1 and 3.5.2) have advantages and disadvantages, which make them best suited to specific situations and/or scenarios.

Comparing the first two methods, it is possible to find some similarities, as both depend on a central entity to control the QoS, which is also responsible for the admission control and therefore for the handover authorization. The differences between them rely on the way the packets get to the new ARs and the selection of the new target AR. One of the large advantages of having a source specific multicast network to support duplicated packets is the previous knowledge of the MTs' surrounding ARs. Also, the multicast groups already assigned allow the MT to move to a finite set of those ARs in the neighbouring, deciding according both to signal and user preferences. Although this is an advantage for the classic fixed network where the ARs are fixed for a long period of time, it is not a good solution for moving ARs, since it is required to continuously update the topology in the QoSB (the creation of the source specific multicast networks requires some time to stabilize). This problem is not present in the first solution, since the MT communicates which ARs it can attach to, and the QoSB decides which of the ARs can handle the MT. However, this process limits the MT choice of ARs, and it is subject to problems in highly dynamic networks where the signal level can change very fast.

Due to the existence of a previous multicast group including the neighbouring routers, the second approach has a large advantage in the handover time. These source/group multicast networks may also be controlled by the QoSB, since it can inform each AR of its multicast network and its surrounding ARs. With this information, the AR can start the join process and establish the multicast networks at boot up (or when it is informed by the QoSB).

In terms of overhead, the second approach has a significant signaling overhead in the wired network. This is due to the existence of a control entity. However, the overhead in the wireless link is low, since the traffic is only inserted in the multicast group upon handover request.

The third approach does not contain a control entity, and therefore, there is no access control and no QoS guarantees in the new network, both for the new flow and for the ones already present in the network. Also, it requires the complete knowledge of the network topology in order to establish the multicast networks. However, this procedure requires very small signaling overhead, and provides a really fast handover without packet loss and additional signaling. In terms of data overhead, all the ARs belonging to the multicast group receive the same data stream, which increases the resource usage in the core network. However, the core network is usually not the bottleneck compared to wireless link. Moreover, this is the only way to grant a continuous stream to wandering MTs. This mechanism is the best suited one for very large mobility scenarios.

3.6 Summary and Conclusions

This chapter proposed three different solutions for handover optimization covering three different scenarios.

The first scenario suits the needs of an operator driven network with no degree of liberty on the choice of the new AR by the MT. The MT always depends on the QoSB decision on the next AP to move. This architecture incorporates several concepts such as candidates’ discovery, context transfer, intelligent interface selection, heterogeneous interfaces, A4C support and, its main purpose, integration of QoS with seamless and fast mobility. The mobility is supported by a QoS subsystem supporting both mobile terminal initiated handovers and network initiated handovers. A mobile terminal initiated handover might be initiated when signal quality is degrading or a new access point is found in the range; at this point the IIS algorithm calculates the best available connection to use, taking into consideration user profile and preferences. If the output of the algorithm is different from the current connection status of the terminal, a handover request is issued. The QoS subsystem, by its turn, checks if the terminal is allowed to move to that target and if the target is capable of providing enough resources for the ongoing sessions of that terminal. Admission control, resource reservations and handover authorization is performed at this point. Packet lost is minimized by the use of duplication and merging mechanisms that duplicates the packets at the old access router by sending the packets directed to the current CoA also to the new CoA and merges them at the terminal. The merging process checks if a packet has already arrived and discards it if a duplicated packet is found. The credentials and security that are allocated at the old AR involve the Context Transfer protocol to transfer that context and activate it at the new AR.

On the other hand, a network initiated handover may be started due to network optimization; the Performance Attendant entity retrieves information about the load of each AP connected to an AR and communicates that information to the Performance Manager that optimizes the network in order to provide the best balanced distribution of terminals regarding user profiles and operator preferences. This operation is only performed when a AR or AP is overloaded or has crossed a defined load threshold.

All these concepts were integrated to provide a complete mobility solution for future operator networks.

In the second proposed solution, a supporting multicast network grants the non predictability of the target AR (in this case a set of neighbouring ARs are prepared to receive the MT). The multicast network allows the reduction of the bandwidth usage inside the operator network assuring the resource optimization, and the delivery of the packets to the surrounding ARs, and thus to the roaming MT. Nevertheless, these two methods (which are operator driven) depend on an entity in the network for handover permission and control. To avoid this in a high mobility network, we suggested a third solution where there is no admission control and where the terminal may have to adapt its flows regarding the available resources of the target ARs.

Chapter 4 Mobility Protocol Implementation

The previous chapter described three architectures to improve handover performance. The first provides seamless and fast handovers in a network with QoS support, where the terminal depends strictly on the network to perform its handovers. The second one relies on a multicast distribution mechanisms that allows the terminal to choose the neighbour access router/access point just before handover ocurs. The third proposed scenario relies on a fast mobility mechanisms with no QoS support adaptad to highly unstable networks or fast moving terminals. Among these three scenarios the architecture that was selected to be implemented and validaded was the first with a mobility mechanism controlled by a central entity using simple duplication of traffic through the new access router. This architecture was integrated in the Daidalos project, and followed the “operator-bias” of this project.

The implementation of the FHO module located at the ARM, responsible for making the translation from ICMPv6 protocol to COPS protocol was done in the scope of this Master Thesis, according to the architecture and the set of modules presented in the previous chapter. Besides this implementation, the architectural design, FHO protocol, messages and parameters definition and validation were also part of this Master Thesis.

This chapter identifies in section 4.1 the required functionalities at the MT and AR to support the mobility architecture as well as its developed modules. A formal description of the mobility protocol integrated with QoS is addressed in section 4.2. It also contains the messages, the interface contents and messages formats. Finally, section 4.3 contains summary and conclusions of this chapter.

4.1 The Developed Software

The development effort of the FHO protocol integrated with QoS was divided by two partners of the Daidalos Project. The implementation of the module in the terminal (FHO Client) and the ICMPv6 Module on the AR was the responsibility of NEC Europe. The translator from ICMPv6 to COPS and interaction with the QoS Broker, denoted as the Fast Handover Module in ARM, was performed in the framework of this MsC thesis. The following sub-sections detail the MT and AR FHO architecture.

4.1.1 Mobile Terminal function

The FHO client architecture is based on one single interface with the MTC. The primitives specified through UNIX sockets allow the exchange of all the required parameters. The module communicates to the network, namely Fast Handover Attendant module in the AR, by means of ICMPv6 messages. The module keeps the information of the handover, implementing its state machine.

A sample debug output of the FHO Client implementation is presented below. As can be seen, it implements the above described mobility protocol.

```
0.000000: got FHO_HO_INITIATE
0.000691: Router Solicitation for Proxy Sent...
0.040268: GOT PrRtAdv
0.040584:FHO_HO_DEC (MIHO) with index 1 sent to MTC
Starting MERGING
0.063505: got FHO_HO_REPORT
0.076120: Sent FBU to FHO:ARM
0.076388: Sent FHO_L2_DISC() to MTC
0.236098: got FHO_L2_CONN()
0.253675: Trying to send FNA.....
0.268158: Trying to send FNA.....
0.284015: Trying to send FNA.....
0.300077: Trying to send FNA.....
0.316016: Trying to send FNA.....
0.332189: Sent FNA to ARM
3.723735: Sent FHO_STOP_MERGING() to MTC
Stopping MERGING
Handover time: 3.723735
```

4.1.2 Access Router function

Figure 32 describes the FHO attendant architecture. The user space daemon has two protocol interfaces (FHO client and QoSB) and two internal local interfaces (D&M agent and Context Transfer). The communication with the FHO client is implemented by means of the ICMPv6 interface, whereas the one with the QoSB is performed via the Advanced Router Manager (ARM) entity. Local communication via UNIX socket allows interaction with the D&M agent and the Context Transfer module.

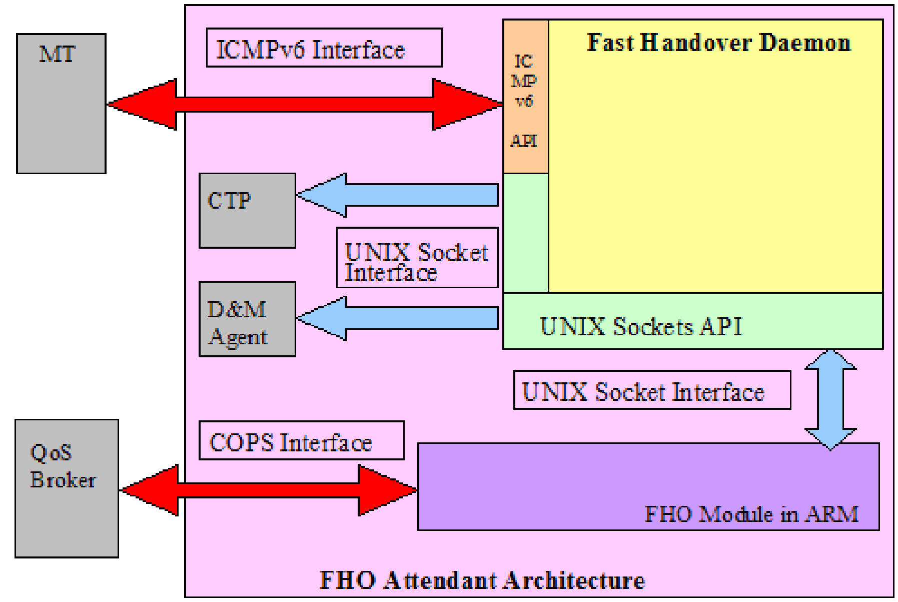


Figure 32 - FHO Attendant Architecture

The FHO Module in the ARM acts as a wrapper (to COPS), of the data collected from the network. It is the module that sends and receives the messages from/to the QoS Broker and handles all of this communication. It interfaces to the daemon using UNIX sockets. The interface between this module and the FHO attendant is done through the UNIX sockets, by sending the ICMPv6 data structure. ICMPv6 messages sent to the MT are built inside the daemon; it retains the information needed to correctly update its state machines. The reverse path is done in the same way: whenever a packet coming from the MT arrives at the AR, the daemon collects the information needed and forwards the packet to the FHO module in the ARM. The advantage of this solution is to have a unique centralized COPS handler in the ARM.

Figure 33 depicts the internal networking functionalities of the FHO attendant module.

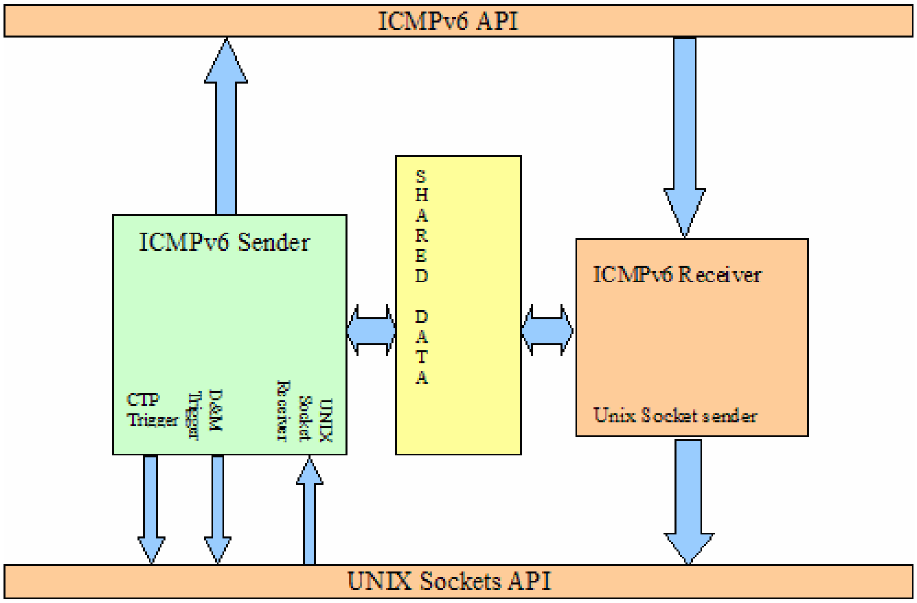


Figure 33 - AR_FHO internal networking functionalities

A sample debug output of the FHO Attendant implementation is presented below. It contains the interaction with the FHO Client.

```
1144085966.703159 : 0.000000 >>>Received RT_SOL_PROXY
Added 2001:690:2380:7776:20a:e4ff:fec0:6377 to hashtable
1144085966.703314 : 0.000155 >>>Sent TriggerHandOverRequest() to ARM
1144085966.737567 : 0.034408 >>>Received HandoverDecision
1144085966.737708 : 0.034549 >>>Sent Trigger_DM( ) to D&M (no)
1144085966.738124 : 0.034965 >>>Sent Pr_Rt_Adv to MT
1144085966.778422 : 0.075263 >>>Received FBU
1144085966.779530 : 0.076371 >>>Sent TriggerHandOverReport() to ARM
1144085967.419663 : 0.716504 >>>Received StopMerging( )
Sleeping 3 sec.......
1144085970.420473 : 3.717314 >>>Sent STOP_DUPLICATING() to D&M
1144085970.420598 : 3.717439 >>>Sent Stop_Merging() to MT
>>>Removed 2001:690:2380:7776:20a:e4ff:fec0:6377 from hashtable
1144085970.420674 >>>Handover time: 3.717515
```

A sample debug output of the FHO@ARM module implementation, which contains all the messages exchanged with the QoS Broker, is the following:

```
   [FHO] 0 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - FHO_ATTENDANT -> ARM : TRIGGER
HANDOVER REQUEST:
   [FHO] 0.000528 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - ARM -> QoS Broker (1) :
HANDOVER REQUEST
   [FHO] 0.031934 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - QoS Broker (1) -> ARM :
DECISION
```

```
   [FHO] 0.032868 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - ARM -> FHO_ATTENDANT :
HANDOVER DECISION

   [FHO] 0.0748801 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - FHO_ATTENDANT -> ARM :
TRIGGER HANDOVER REPORT: SUCCESS

   [FHO] 0.0753700 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - ARM -> QoS Broker (1) :
HANDOVER REPORT

   [FHO] 0.7136490 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - QoS Broker (1) -> ARM :
DECISION

   [FHO] 0.7145490 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) -  ARM -> FHO_ATTENDANT : STOP
MERGING

   [FHO] 0.714913 (2001:690:2380:7776:20a:e4ff:fec0:6377,3) - ARM -> QoS Broker (1) :
HANDOVER REPORT

   [FHO] 0.714913 Handover Time
```

4.2 Protocol Description

In this section it is presented a description of the messages of the mobility protocol. This description includes the mobility messages between the MT and the AR and between the AR and the QoS Broker and their specification and contents.

4.2.1 Interface Contents

At the handover request the terminal sends up to three candidates ARs to the network. This avoids the need to issue additional handover signalling if a first request is refused and more possible targets are available. By using this list of candidates, containing the link layer address of the access point, the access router IPv6 address and the CoA that the terminal will use in that access router (ordered by preference), the network is able to choose the first one matching the criteria for authorization an handover (service level agreements, QoS resources on the new attachment point, authorization, etc.), and answers with the selected attachment point; it may also deny the handover if none of the targets matches the criteria.

If the handover is denied, the network includes an Handover Refused Option which contains an explanation of the denial; based on that, the terminal can act accordingly for future handovers. For instance, if the handover was denied due to authorization, the terminal already knows that it is not authorized to use that attachment point and will not request another handover to that target. Otherwise, if the handover is accepted, the network will answer with the Accepted AR Option where it will indicate which of the candidates was selected.

The network has the ability to instruct the terminal to move to a determined target even without its request. The information on the new target is conveyed by the Candidate AR Option; in this case, only one candidate is present in the message and the terminal must move to the new target.

All ICMPv6 messages presented in this protocol require a generic header (ICMPv6/FHO_HEADER) which contains the basic characteristics of this handover, such as the message type, handover session and the identification of the terminal.

The developed mobility protocol has two underlaying protocols for the transport of the contents. From the terminal to the network, the messages are conveyed by means of ICMPv6 [RFC2463] messages. This protocol was chosen due to its controller nature. The QoS subsystem relied on COPS [RFC2748] protocol to exchange messages among its entities. Since the need for integration of the mobility and QoS part was a requirement, a translator between ICMPv6 and COPS messages was implemented. Both protocols were extended to support the mobility signalling, and the messages were defined similarly (when possible) for both protocols.

Table 1 includes the description of the ICMPv6 messages exchanged by the FHO in the MT (MT_FHO) and the AR (AR_FHO), depicted at Figure 34. The tables include the direction of each message represented by the source and the destination module, the primitive which is referring to, and the fields that are sent in the messages. There are fields that are mandatory and others optional; the optional fields are represented between straight brackets (i.e. [<Link Layer Address Option>]).

Some options were included to provide more consistency and inteligence to the network. The Handover Priority Option plays an important role on prioritization, since it contains the information about the imminency of signal loss. It means that when the signal is getting worse very fast, the software needs to realize that the connection can break at any moment: this option is used to indicate this fact to the network to give full priority to this handover. With this flag set to immiment, the decision of handover is taken as soon as possible, as well as the QoS resources reservation at the new attachment point. This means that imminent handovers have more priority in the internal queueing of messages, which allows them to be treated as fast as possible. Handovers that are not imminent are also taken care of as fast as possible but might be delayed due to imminent handovers higher priority.

Section 4.2.2.1 details the messages contents.

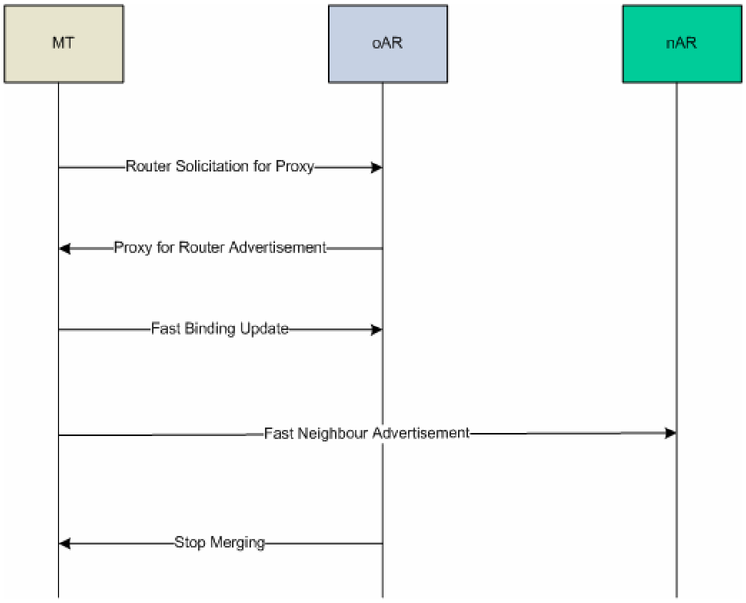

MT
oAR
nAR
Router Solicitation for Proxy
Proxy for Router Advertisement
Fast Binding Update
Fast Neighbour Advertisement
Stop Merging

Figure 34 - Mobile Terminal Initiated Handover message MT<->AR_FHO

Source	Destination	Primitive	Message Contents
MT_FHO	AR_FHO	RtSolPr	<ICMPv6/FHO Header> <Handover Priority Option> <Link Layer Address Option> <IPv6 Address Option> (oCoA) <Candidate AR Option> <Link Layer Address Option> (Candidate LLA 1) <IPv6 Address Option> (Candidate AR IPv6 1) <IPv6 Address Option> (Candidate nCoA 1) [<Link Layer Address Option>] (Candidate LLA 2) [<IPv6 Address Option>] (Candidate AR IPv6 2) [<IPv6 Address Option>] (Candidate nCoA 2) [<Link Layer Address Option>] (Candidate LLA 3) [<IPv6 Address Option>] (Candidate AR IPv6 3) [<IPv6 Address Option>] (Candidate nCoA 3)
AR_FHO	MT_FHO	PrRtAdv	<ICMPv6/FHO Header> <Link Layer Address Option> [<Handover Refused Option>] [<Accepted AR Option>] [<Candidate AR Option>] [<Link Layer Address Option>] (new AP LLA) [<IPv6 Address Option>] (new AR IPv6) [<IPv6 Address Option>] (nCoA)

MT_FHO	AR_FHO	FBU	<ICMPv6/FHO Header>
MT_FHO	AR_FHO	FNA	<ICMPv6/FHO Header> <Link Layer Address Option>
AR_FHO	MT_FHO	Stop Merging	<ICMPv6/FHO Header>

Table 1 - Primitives of Interface MT_FHO <-> AR_FHO

Figure 35 shows the signaling flow for the MIHO scenario depicting the message exchange between the Fast Handover Module in the AR (AR_FHO) and the QoS Broker (QoSB). Since we've used the COPS protocol and model to support this communication, the figures depict a PDP (Policy Decision Point) which represents QoSB and two PEPs (Policy Enforcement Poin) representing the old and the new AR.

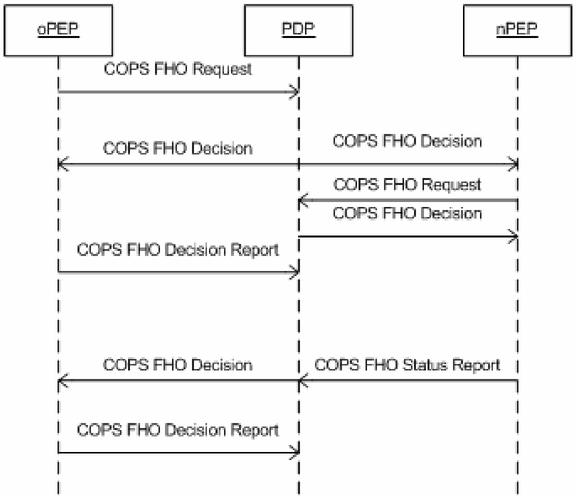


Figure 35 – Mobile Terminal Initiated Handover message QoS<->AR_FHO

Table 2 includes the description of the COPS messages depicted above. The Common Header, Client Handle, Context, Client SI, "Decision: Flags" and Decision Client SI messages are specified in the COPS standard and are used with some minor adjustments to fields codes in order to support Daidalos extensions. Making a parallel with the ICMPv6 messages previously presented it is possible to understand that the COPS Daidalos FHO ID is similar to the ICMPv6 FHO Header message and the same happens to the other objects. Section 4.2.2.2 presents further detail on COPS messages and formats.

Source	Destination	Primitive	Message Contents
AR_FHO	QoSB	HandoverRequest	<Common Header> <Client Handle> <Context> <Client SI (s)> <Daidalos FHO ID> <Daidalos Candidate AR Object> [<Daidalos Candidate AR Object>] [<Daidalos Candidate AR Object>]
QoSB	AR_FHO	HandoverDecison	<Common Header> <Client Handle> <Context> <Decision: Flags> <Decision: Client SI (s)> <Daidalos FHO ID> [<Daidalos Accepted AR>] [<Daidalos FHO Status>]
AR_FHO	QoSB	HandoverReport	<Common Header> <Client Handle> <Report Type> <Client SI> [<Daidalos FHO Status>]

Table 2 - Primitives of Interface AR_FHO <-> QoSB for MIHO

For the NIHO scenario, Figure 36 depicts the signaling and Table 3 describes the COPS messages contents. The contents are basically the same as in MIHO scenario, although the COPS FHO Request that triggers the process is missing. The COPS FHO Report received after the COPS FHO Decision in the MIHO scenario is replaced by a COPS FHO Request in order to comply with the COPS standards about naming convention. The objects used in these messages are the same that were used in the MIHO scenario.

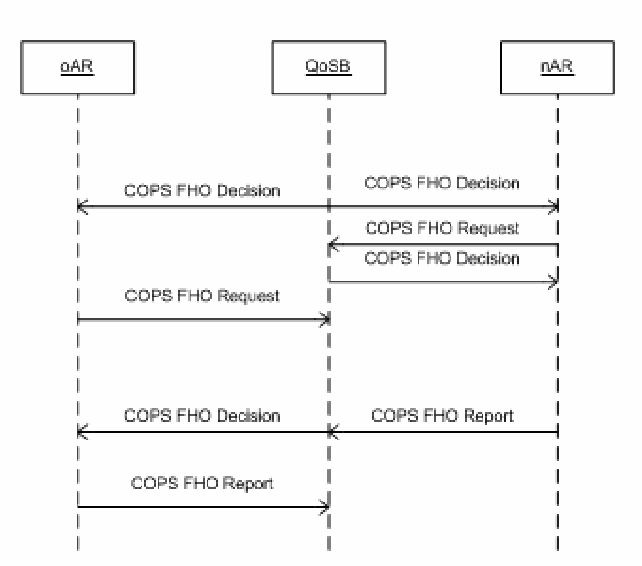


Figure 36 – Network Initiated Handover message QoS<->AR_FHO

Source	Destination	Primitive	Parameters
AR_FHO	QoSB	HandoverRequest	<Common Header> <Client Handle> <Context> <Client SI (s)> <Daidalos FHO ID> <Daidalos Candidate AR Object> [<Daidalos Candidate AR Object>] [<Daidalos Candidate AR Object>]
QoSB	AR_FHO	HandoverDecison	<Common Header> <Client Handle> <Context> <Decision: Flags> <Decision: Client SI (s)> <Daidalos FHO ID> [<Daidalos Candidate AR>] [<Daidalos FHO Status>]
AR_FHO	QoSB	HandoverReport	<Common Header> <Client Handle> <Report Type> <Client SI> [<Daidalos FHO Status>]

Table 3 - Primitives of Interface AR_FHO <-> QoSB for NIHO

4.2.2 Messages Format

In this section we describe the different messages and options types. We consider both ICMPv6 messages between the MT and the AR, and COPS messages between the AR and QoSB.

4.2.2.1 Format of ICMPv6 messages

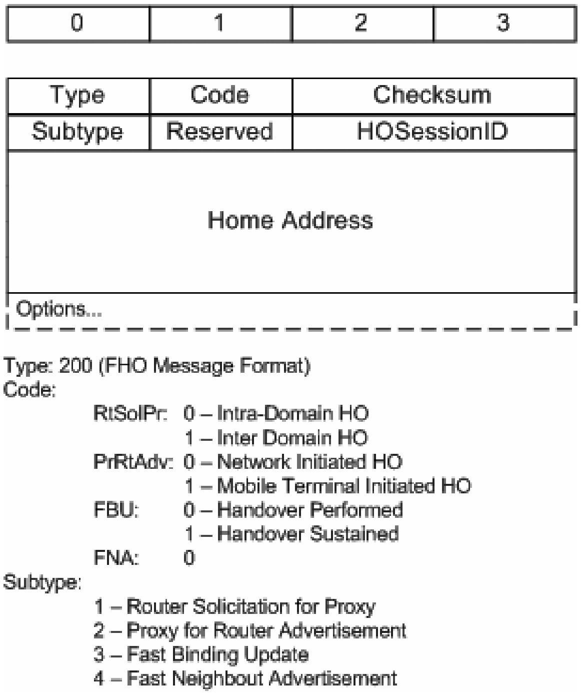
This section presents the definition of ICMPv6 messages format and contents that are part of the presented protocol.

4.2.2.1.1 ICMPv6/FHO Header

To use ICMPv6 messages, we need to comply with the standard which refers the use of 4 bytes aligned structures and Type Length Value (TLV) message format. The ICMPv6 messages used begin with the field Type that determines the type of message being addressed (Figure 37). In this protocol we used Type 200 to indicate FHO Protocol message. The value of the field code is dependent of the subtype with different meaning per code.

The reserved fields along these messages represent a block of bytes set to 0 in order to keep the structure 4 bytes aligned. The Handover Session ID is a unique number for each handover that represents and relates the messages of a single handover procedure. The Home Address/VID is a unique identifier of the terminal; when used together with the Session ID, it is possible to univocally identify a handover of a specific terminal.

This header is included in each ICMPv6 message that is exchanged between the MT and the AR; the options are added at the end of this header.

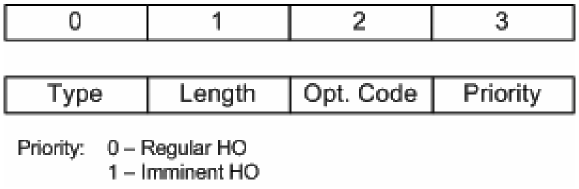


Type: 200 (FHO Message Format)
Code:
RtSolPr: 0 – Intra-Domain HO
1 – Inter Domain HO
PrRtAdv: 0 – Network Initiated HO
1 – Mobile Terminal Initiated HO
FBU: 0 – Handover Performed
1 – Handover Sustained
FNA: 0
Subtype:
1 – Router Solicitation for Proxy
2 – Proxy for Router Advertisement
3 – Fast Binding Update
4 – Fast Neighbout Advertisement

Figure 37 - ICMPv6 FHO Header

4.2.2.1.2 Handover Priority Option

This option, depicted in Figure 38, represents the priority of the handover by informing the network if it is a regular handover, or if it is imminent and this request is to be treated with extreme urgency.



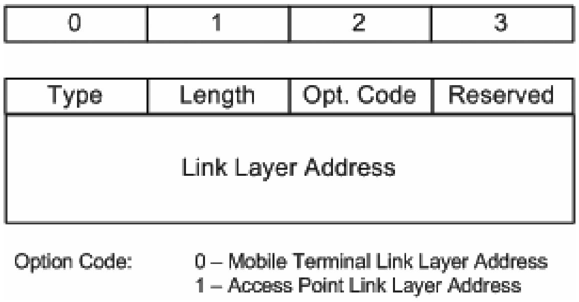
0	1	2	3
Type	Length	Opt. Code	Priority

Priority: 0 – Regular HO
1 – Imminent HO

Figure 38 - ICMPv6 FHO Priority Option

4.2.2.1.3 Link-Layer Address (LLA) Option

This option, depicted in Figure 39, represents the Link Layer Address of the moving terminal or the target AP dependent on the option code set. The Link Layer Address field contains the address itself.



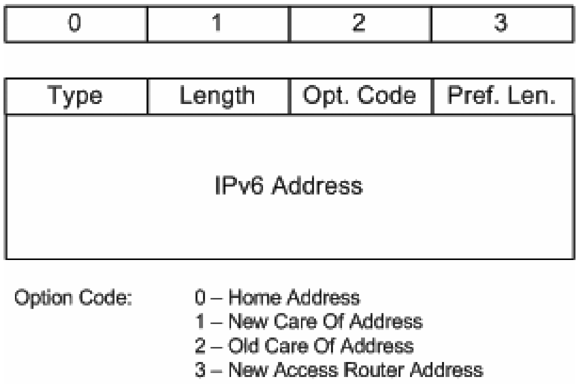
0	1	2	3
Type	Length	Opt. Code	Reserved
Link Layer Address			

Option Code: 0 – Mobile Terminal Link Layer Address
1 – Access Point Link Layer Address

Figure 39 - ICMPv6 FHO LLA Option

4.2.2.1.4 IPv6 Address Option

As in the above option, the option code determines the representation of the IPv6 Address that is included in the field with the same name. This option code can indicate if this field refers to a HoA, new CoA to be used, the old CoA that was in use or the new AR address. This option is depicted in Figure 40.



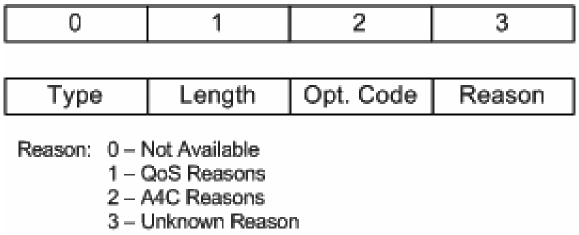
0	1	2	3
Type	Length	Opt. Code	Pref. Len.
IPv6 Address			

Option Code: 0 – Home Address
1 – New Care Of Address
2 – Old Care Of Address
3 – New Access Router Address

Figure 40 - ICMPv6 FHO IPv6 Address Option

4.2.2.1.5 Handover Refused Option

This option, depicted in Figure 41, is only used whenever the header code is equal to 2 and the subtype is equal to 2. This indicates a handover refusal and provides the reason for that action, which can be Not Available/Provided, QoS reasons, A4C reasons or Unknown reason (this latter code value is mainly used for inter-AN scenarios where the operators may try to hide this information to other users because of policy reasons).



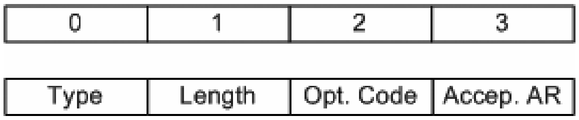
0	1	2	3
Type	Length	Opt. Code	Reason

Reason: 0 – Not Available
1 – QoS Reasons
2 – A4C Reasons
3 – Unknown Reason

Figure 41 - ICMPv6 FHO Refused Option

4.2.2.1.6 Accepted AR Option

Whenever the code is different from 2 but the subtype is equal to 2, then the handover was accepted and the accepted AR is informed. This field contains the number of order of the candidate sent on the request message. For instance, if the selected AR was the first candidate, this field is set to 0; if it was the second candidate, it is set to 1 and so on. This option is depicted in Figure 42.

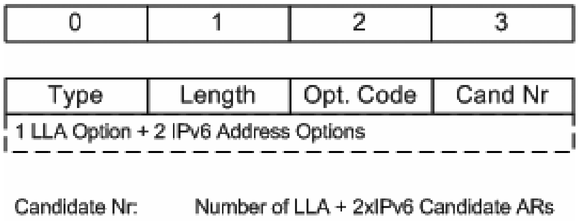


0	1	2	3
Type	Length	Opt. Code	Accep. AR

Figure 42 - ICMPv6 FHO Accepted AR Option

4.2.2.1.7 Candidate AR Option

The candidate AR option, depicted in Figure 43, is sent on a NIHO Proxy for Router Advertisement message or on a MIHO Router Solicitation for Proxy message. This includes the number of candidates provided to the network (up to three); in the case of NIHO, the Candidate number field is set to 1. Each candidate must be defined by one Link Layer Address Option representing the target AP, and two IPv6 Address Options representing the new AR and the new CoA. The new CoA to be used in the new network is here provided in order to detect duplicate addresses before the handover occurs, saving the time spent on duplicate address detection mechanisms [RFC4429] after connecting to the new link.



Candidate Nr: Number of LLA + 2xIPv6 Candidate ARs

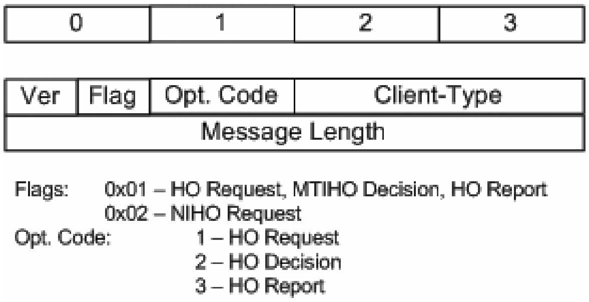
Figure 43 - ICMPv6 FHO Candidate AR Option

4.2.2.2 Format of COPS messages

The COPS protocol was extended to support Daidalos COPS messages including FHO signaling. In the following sections we describe the COPS objects that were modified and how they translate to ICMPv6 TLVs. Standard objects are also presented because of COPS messages construction requirements, which forced to include some objects that do not require extensions.

4.2.2.2.1 Common Header

The common header, depicted in Figure 44, is present in every COPS messages exchanged. It indicates the type of message (Report, Decision or Report), the type of handover (MIHO or NIHO) and contains the total message length.

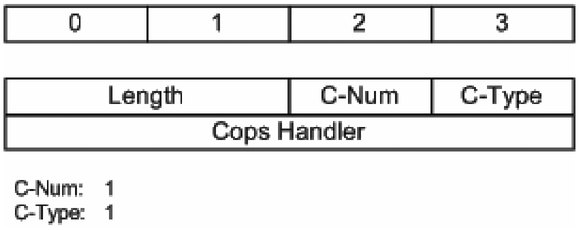


Flags: 0x01 – HO Request, MTIHO Decision, HO Report
0x02 – NIHO Request
Opt. Code: 1 – HO Request
2 – HO Decision
3 – HO Report

Figure 44 - COPS Common Header

4.2.2.2.2 Client Handle

Client handle, depicted in Figure 45, is also present in all messages; it is used for COPS protocol to uniquely identify the COPS session of each message.

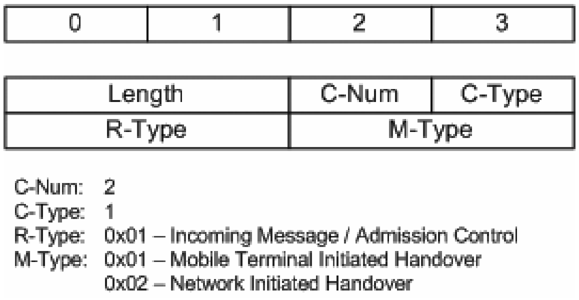


C-Num: 1
C-Type: 1

Figure 45 - COPS Client Handle

4.2.2.2.3 Context

Due to COPS requirements, requests and decision messages must include this object, depicted in Figure 46. It can indicate the type of handover, although this information is redundant as it was already inserted in the common header object.



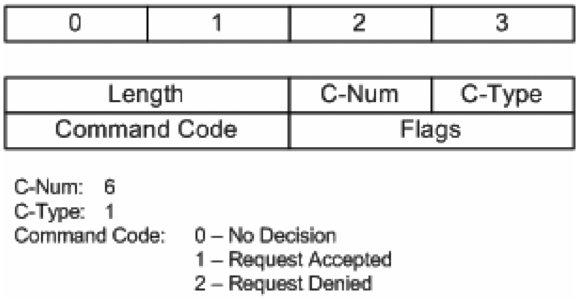
0	1	2	3
Length		C-Num	C-Type
R-Type		M-Type	

C-Num: 2
C-Type: 1
R-Type: 0x01 – Incoming Message / Admission Control
M-Type: 0x01 – Mobile Terminal Initiated Handover
0x02 – Network Initiated Handover

Figure 46 - COPS Context

4.2.2.2.4 Decision: Flags

This object, depicted in Figure 47, is included in every decision and informs about the success of the handover by setting the command code to 1 or 2, to indicate that the handover was accepted or denied, respectively. Whenever this messages refers to a decision to convey information about the correct attachment of the terminal at the new AP, the command code is set to 0.



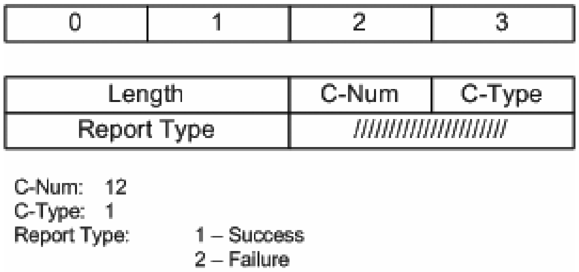
0	1	2	3
Length		C-Num	C-Type
Command Code		Flags	

C-Num: 6
C-Type: 1
Command Code: 0 – No Decision
1 – Request Accepted
2 – Request Denied

Figure 47 - COPS Decision Flags

4.2.2.2.5 Report-Type

The report type object, depicted in Figure 48, is used for reporting purposes indicating the success or failure of the handover. It is used to inform the QoS Broker about the terminal attachment to the new network, and about the confirmation that the terminal is performing handover. This latest message is sent after the reception of the Fast Binding Update message for a mobile initiated handover.



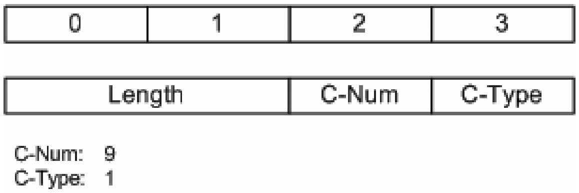
0	1	2	3
Length		C-Num	C-Type
Report Type		/////////////////////	

C-Num: 12
C-Type: 1
Report Type: 1 – Success
2 – Failure

Figure 48 - COPS Report Type

4.2.2.2.6 ClientSI Message Header

The Client Specific Information object, depicted in Figure 49, is the regular Client SI Header from the COPS Protocol. It acts as a container/header for other COPS objects containing client specific information.



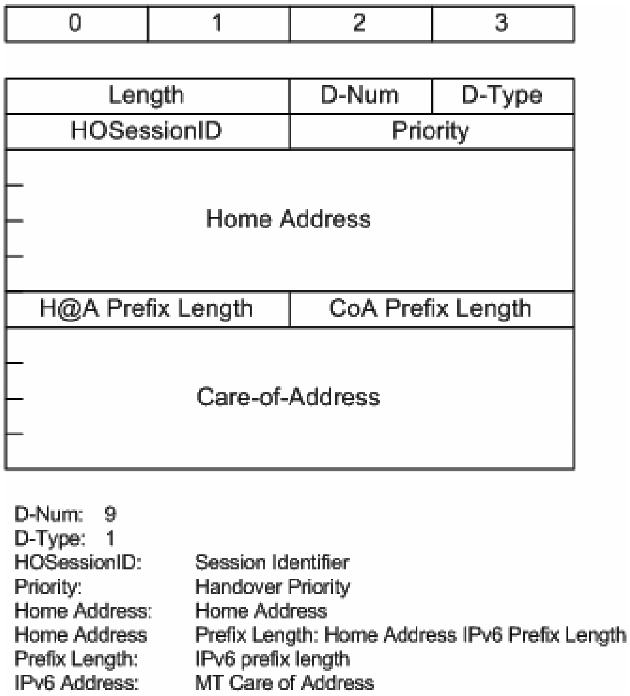
0	1	2	3
Length		C-Num	C-Type

C-Num: 9
C-Type: 1

Figure 49 - COPS Client SI

4.2.2.2.7 Daidalos FHO ID

This message resembles the ICMPv6 FHO Header. It is included in every message and refers to the ICMPv6 FHO header for field information. This message, depicted in Figure 50, aggregates some of the TLVs that were always present on the ICMPv6 messages, such as the IPv6 address representing the old (current) CoA of the terminal.



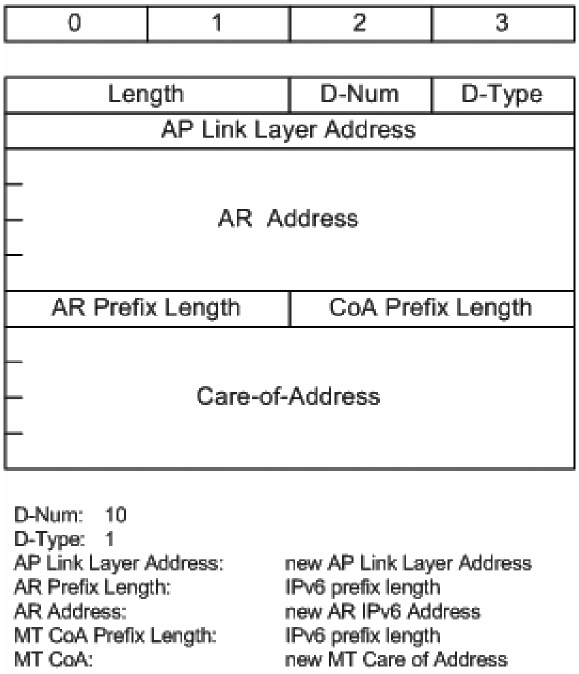
D-Num: 9
D-Type: 1

HOSessionID:	Session Identifier
Priority:	Handover Priority
Home Address:	Home Address
Home Address	Prefix Length: Home Address IPv6 Prefix Length
Prefix Length:	IPv6 prefix length
IPv6 Address:	MT Care of Address

Figure 50 - COPS Daidalos FHO ID

4.2.2.2.8 Daidalos FHO Candidate AR Object

This candidate AR object, depicted in Figure 51, is similar to the ICMPv6 one. However, whereas the ICMPv6 has a TLV for each of the items, this object aggregates all the options into this header.



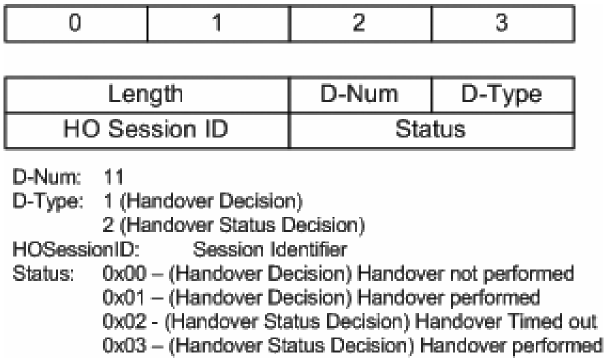
D-Num: 10
D-Type: 1

AP Link Layer Address:	new AP Link Layer Address
AR Prefix Length:	IPv6 prefix length
AR Address:	new AR IPv6 Address
MT CoA Prefix Length:	IPv6 prefix length
MT CoA:	new MT Care of Address

Figure 51 - COPS Daidalos Candidate AR

4.2.2.2.9 Daidalos FHO Status Object

This object, depicted in Figure 52, is included in reports and some decision messages. It indicates the HO Session ID and the status of the handover identified by that ID.



0	1	2	3

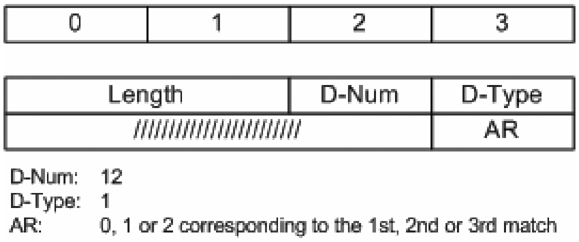
Length		D-Num	D-Type
HO Session ID		Status	

D-Num: 11
D-Type: 1 (Handover Decision)
2 (Handover Status Decision)
HOSessionID: Session Identifier
Status: 0x00 – (Handover Decision) Handover not performed
0x01 – (Handover Decision) Handover performed
0x02 - (Handover Status Decision) Handover Timed out
0x03 – (Handover Status Decision) Handover performed

Figure 52 - COPS Daidalos FHO Status

4.2.2.2.10 Daidalos FHO Accepted AR Object

Similar to the ICMPv6 Accepted AR TLV this object, depicted in Figure 53, has the same functionality.



0	1	2	3

Length		D-Num	D-Type
//////////////////////			AR

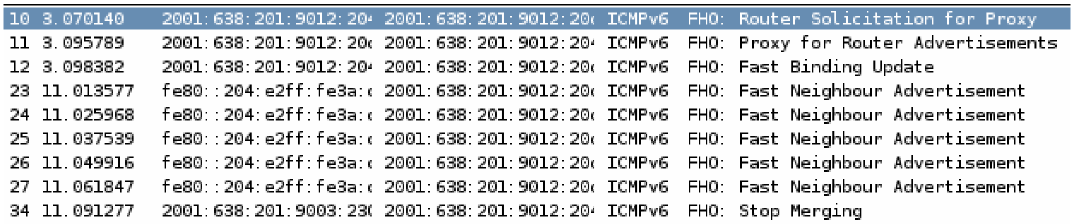
D-Num: 12
D-Type: 1
AR: 0, 1 or 2 corresponding to the 1st, 2nd or 3rd match

Figure 53 - COPS Daidalos FHO Accepted AR

4.2.2.3 Ethereal dissector

Ethereal [ethereal] is a packet analyzer that enables the user to check the contents of each packet flowing on the network. In the framework of this MsC thesis, a ethereal dissector was developed which allowed a better view and a correct analysis of the packets of this protocol. The dissector is working for the ICMPv6 FHO protocol messages.

The following figures show the ICMPv6 FHO protocol messages, inside Ethereal.



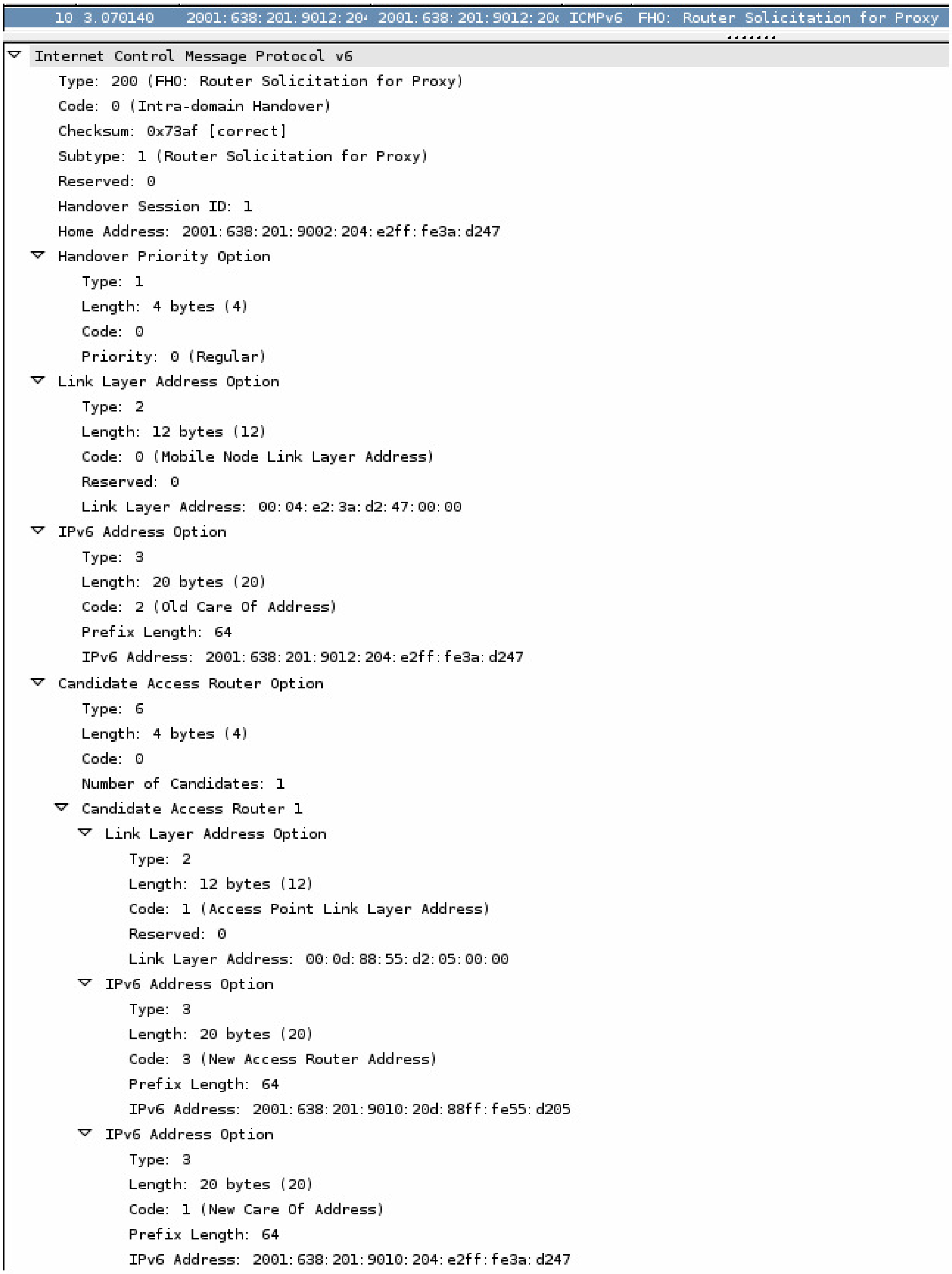
```
10 3.070140    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Router Solicitation for Proxy
11 3.095789    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Proxy for Router Advertisements
12 3.098382    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Fast Binding Update
23 11.013577   fe80::204:e2ff:fe3a: 2001:638:201:9012:20 ICMPv6  FHO: Fast Neighbour Advertisement
24 11.025968   fe80::204:e2ff:fe3a: 2001:638:201:9012:20 ICMPv6  FHO: Fast Neighbour Advertisement
25 11.037539   fe80::204:e2ff:fe3a: 2001:638:201:9012:20 ICMPv6  FHO: Fast Neighbour Advertisement
26 11.049916   fe80::204:e2ff:fe3a: 2001:638:201:9012:20 ICMPv6  FHO: Fast Neighbour Advertisement
27 11.061847   fe80::204:e2ff:fe3a: 2001:638:201:9012:20 ICMPv6  FHO: Fast Neighbour Advertisement
34 11.091277   2001:638:201:9003:23 2001:638:201:9012:20 ICMPv6  FHO: Stop Merging
```

Figure 54 - Message exchange

As an example, Figure 55, represents a Router Solicitation for Proxy message. This message is identified as a Router Solicitation for Proxy message due to the fact that it is of Type 200 – FHO Protocol and of subtype 1. The Code 0 indicates that it is an interdomain handover. The session identifier for this handover is 1 and the home address of the terminal is 2001:638:201:9002:204:e2ff:fe3a:d247. This message includes 4 options, a handover priority option, a link layer address option, a IPv6 address option and a candidates access router option containing 3 candidates options.

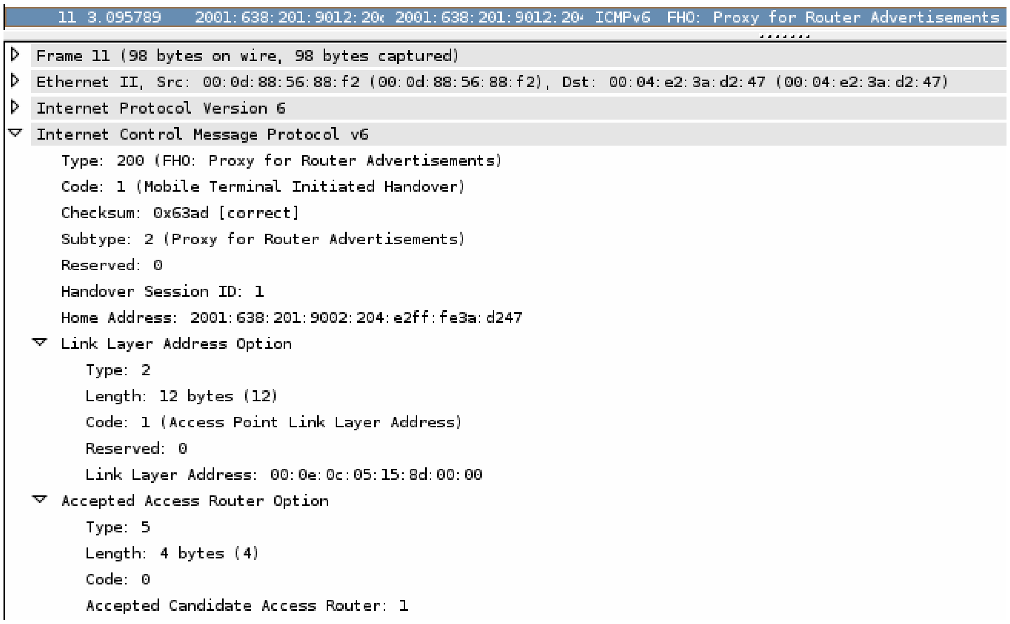
The handover priority option indicated by the option type 1 informs about the priority of the handover and in this case it is a regular handover (0) meaning that there's no imminence of connection lost. This option is then followed by the link layer address option (type 2) which contains the link layer address (00: 04:e2:3a:d2:47) of the terminal interface (code 0). In case of multi interface terminals, this option identifies clearly which of the interfaces is about to move. The IPv6 address option (type 3) carries the information of the interface's current CoA (code 2) concerning the IPv6 Address Prefix, 64, and the IPv6 address 2001:638:201:9012:204:e2ff:fe3a:d247.

The candidate access router option (code 6) contains the information about the candidates in this case it contains only one option. Each option consists on a set of three elements, a link layer address (type 2) of the access point (code 1) which is 00:0d:88:55:d2:05, an IPv6 address (type 3) representing the address of the new access router (code 3) with its prefix length (64) and its address (2001:638:201:9010:20d:88ff:fe55:d205), and another IPv6 address (type 3) containing the CoA (code 1) that the terminal will use (2001:638:201:9010:204:e2ff:fe3a:d247) when connected to the new network.



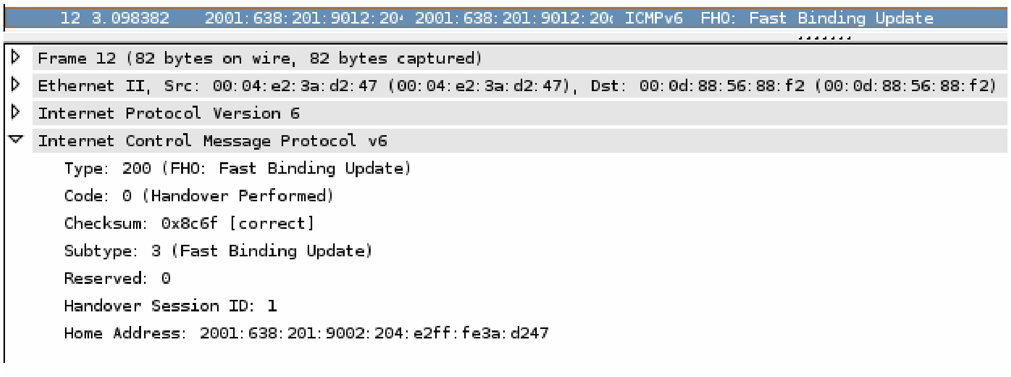
```
     10 3.070140    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Router Solicitation for Proxy

▽ Internet Control Message Protocol v6
    Type: 200 (FHO: Router Solicitation for Proxy)
    Code: 0 (Intra-domain Handover)
    Checksum: 0x73af [correct]
    Subtype: 1 (Router Solicitation for Proxy)
    Reserved: 0
    Handover Session ID: 1
    Home Address: 2001:638:201:9002:204:e2ff:fe3a:d247
  ▽ Handover Priority Option
      Type: 1
      Length: 4 bytes (4)
      Code: 0
      Priority: 0 (Regular)
  ▽ Link Layer Address Option
      Type: 2
      Length: 12 bytes (12)
      Code: 0 (Mobile Node Link Layer Address)
      Reserved: 0
      Link Layer Address: 00:04:e2:3a:d2:47:00:00
  ▽ IPv6 Address Option
      Type: 3
      Length: 20 bytes (20)
      Code: 2 (Old Care Of Address)
      Prefix Length: 64
      IPv6 Address: 2001:638:201:9012:204:e2ff:fe3a:d247
  ▽ Candidate Access Router Option
      Type: 6
      Length: 4 bytes (4)
      Code: 0
      Number of Candidates: 1
    ▽ Candidate Access Router 1
      ▽ Link Layer Address Option
          Type: 2
          Length: 12 bytes (12)
          Code: 1 (Access Point Link Layer Address)
          Reserved: 0
          Link Layer Address: 00:0d:88:55:d2:05:00:00
      ▽ IPv6 Address Option
          Type: 3
          Length: 20 bytes (20)
          Code: 3 (New Access Router Address)
          Prefix Length: 64
          IPv6 Address: 2001:638:201:9010:20d:88ff:fe55:d205
      ▽ IPv6 Address Option
          Type: 3
          Length: 20 bytes (20)
          Code: 1 (New Care Of Address)
          Prefix Length: 64
          IPv6 Address: 2001:638:201:9010:204:e2ff:fe3a:d247
```

Figure 55 – Router Solicitation For Proxy Message



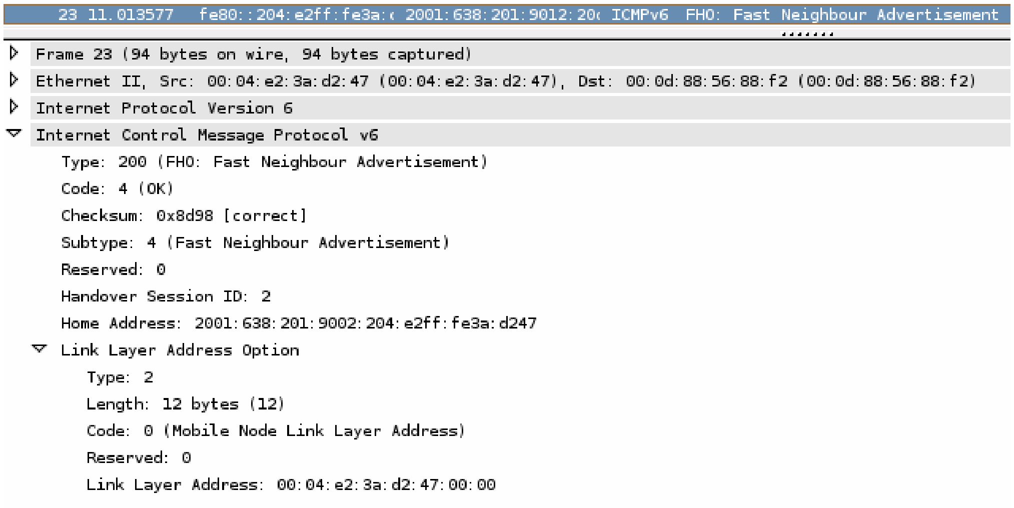
```
   11 3.095789    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Proxy for Router Advertisements
▷ Frame 11 (98 bytes on wire, 98 bytes captured)
▷ Ethernet II, Src: 00:0d:88:56:88:f2 (00:0d:88:56:88:f2), Dst: 00:04:e2:3a:d2:47 (00:04:e2:3a:d2:47)
▷ Internet Protocol Version 6
▽ Internet Control Message Protocol v6
    Type: 200 (FHO: Proxy for Router Advertisements)
    Code: 1 (Mobile Terminal Initiated Handover)
    Checksum: 0x63ad [correct]
    Subtype: 2 (Proxy for Router Advertisements)
    Reserved: 0
    Handover Session ID: 1
    Home Address: 2001:638:201:9002:204:e2ff:fe3a:d247
  ▽ Link Layer Address Option
      Type: 2
      Length: 12 bytes (12)
      Code: 1 (Access Point Link Layer Address)
      Reserved: 0
      Link Layer Address: 00:0e:0c:05:15:8d:00:00
  ▽ Accepted Access Router Option
      Type: 5
      Length: 4 bytes (4)
      Code: 0
      Accepted Candidate Access Router: 1
```

Figure 56 - Proxy for Router Advertisements Message



```
   12 3.098382    2001:638:201:9012:20 2001:638:201:9012:20 ICMPv6  FHO: Fast Binding Update
▷ Frame 12 (82 bytes on wire, 82 bytes captured)
▷ Ethernet II, Src: 00:04:e2:3a:d2:47 (00:04:e2:3a:d2:47), Dst: 00:0d:88:56:88:f2 (00:0d:88:56:88:f2)
▷ Internet Protocol Version 6
▽ Internet Control Message Protocol v6
    Type: 200 (FHO: Fast Binding Update)
    Code: 0 (Handover Performed)
    Checksum: 0x8c6f [correct]
    Subtype: 3 (Fast Binding Update)
    Reserved: 0
    Handover Session ID: 1
    Home Address: 2001:638:201:9002:204:e2ff:fe3a:d247
```

Figure 57 - Fast Binding Update Message



```
   23 11.013577   fe80::204:e2ff:fe3a:( 2001:638:201:9012:20( ICMPv6  FHO: Fast Neighbour Advertisement
▷ Frame 23 (94 bytes on wire, 94 bytes captured)
▷ Ethernet II, Src: 00:04:e2:3a:d2:47 (00:04:e2:3a:d2:47), Dst: 00:0d:88:56:88:f2 (00:0d:88:56:88:f2)
▷ Internet Protocol Version 6
▽ Internet Control Message Protocol v6
    Type: 200 (FHO: Fast Neighbour Advertisement)
    Code: 4 (OK)
    Checksum: 0x8d98 [correct]
    Subtype: 4 (Fast Neighbour Advertisement)
    Reserved: 0
    Handover Session ID: 2
    Home Address: 2001:638:201:9002:204:e2ff:fe3a:d247
  ▽ Link Layer Address Option
      Type: 2
      Length: 12 bytes (12)
      Code: 0 (Mobile Node Link Layer Address)
      Reserved: 0
      Link Layer Address: 00:04:e2:3a:d2:47:00:00
```

Figure 58 – Fast Neighbour Advertisement

4.3 Summary and Conclusions

This chapter presented a description of the developed modules, interfaces and protocols, including a detailed specification of the message formats and its objects. This specification resulted on the implementation of some modules of the Fast Handover Protocol, namely the FHO Client located at the MT, the FHO Attendant located at the AR and FHO@ARM located at the AR module ARM. The communication between FHO Client and FHO Attendant is performed using the FHO protocol which runs on ICMPv6 and then forwards the messages to FHO@ARM through UNIX Sockets. As part of the specification and architecture the COPS protocol was extended to support mobility procedures thus integrating mobility and QoS signalling to provide an integrated architecture. The extended COPS protocol is used between the AR and the QoS Broker.

In the end, the implemented functionalities were tested and validated with the developed Ethereal dissector. This dissector helped to identify some misconfigurations, internal module errors and wrong values passed between modules, while providing an easy way of visualizing the protocol.

Chapter 5 Mobility Performance Measurements

In this MsC Thesis a mobility architecture integrating QoS and mobility mechanisms to support QoS and real-time requirements in an operator-driven network is proposed. To aid in the efficiency of mobility, two mobility architectures that make use of multicast networks to distribute the traffic among the surrounding access routers were also proposed. To evaluate the proposed mobility protocols, it is presented in Section 5.1 the effort to implement the protocols in the OMNeT++ simulator.

To evaluate the Daidalos mobility architecture, the several mobility and QoS modules were implemented and integrated in a real demonstrator. Section 5.2 presents the integration efforts performed to implement mobility and QoS in a real testbed, the tests conducted to the demonstrator, and the results obtained in terms of the performance of the proposed architecture. This performance is evaluated concerning intra- and inter-techonology handovers. The specifc measurements addressed are handover timings, packet losses, delays and jitter. The impacts on both TCP and UDP communications are also evaluated in section 5.3 where it is showed that this architecture is able to assure fast handover of mobile terminals with ongoing communications, with small handover timings during handover. Finally, this chapter also describes some limitations found in the implementation and in the architecture.

5.1 Simulating Daidalos Mobility

This section presents the effort conducted in terms of the simulation of the proposed mechanisms. The simulator chosen was OMNeT++ [omnet] due to its modularity and because at that time it claimed to implement a Mobile IPv6 stack (which this work was based on). It is designed for computer network simulations. Other simulators considered [ns2][opnet] didn’t have native support for mobile ipv6 and the most recent implementations were outdated and didn’t support route optimization [mobiwan].

5.1.1 The Omnet++ Simulator

OMNeT++ stands for Objective Modular Network Testbed in C++**.** It is a discrete event simulation tool designed to simulate computer networks, multi-processors and other distributed systems. Its applications can be extended for modeling other systems as well. It has become a popular network simulation tool in the scientific community, as well as in industry over the years. The principal author is András Varga, with occasional contributions from a number of people.

OMNeT++ is free for any non-profit use. The author must be contacted if it is used in a commercial project.

The components of OMNeT++ are the following:

- Simulation kernel library;
- Compiler for the Network Description (NED) topology description language (nedc);
- Graphical network editor for NED files (GNED);
- GUI for simulation execution, links into simulation executable (Tkenv);
- Command-line user interface for simulation execution (Cmdenv);
- Graphical output vector plotting tool (Plove);
- Utilities (random number seed generation tool, makefile creation tool, etc.);
- Documentation, sample simulations, contributed material, etc.

OMNeT++ works well on multiple platforms. It was first developed on Linux. Omnet++ runs on most Unix systems and Windows platforms (it works better on NT4.0, W2K or XP).

The best platforms to use are:

- Solaris, Linux (or other Unix-like systems) with GNU tools;
- Win32 and Cygwin32 (Win32 port of gcc);
- Win32 and Microsoft Visual C++.

5.1.1.1 Simulation Modeling in Omnet++

The following types of modeling can be used:

- Communication protocols;
- Computer networks and traffic modeling;
- Multi-processor and distributed systems;
- Administrative systems;
- Any other system where the discrete event approach is suitable.

This flexibility is supported by adequate libraries.

Object libraries can be made using simple modules. The best simple modules to be used for library modules are the ones that implement:

- Physical/Data-link protocols: Ethernet, Token Ring, FDDI, LAPB etc;
- Higher layer protocols: IP,IPv6, MIPv6, TCP, X.25 L2/L3, etc;

- Network application types: E-mail, NFS, X, audio etc;
- Basic elements: message generator, sink, concentrator/simple hub, queue etc;
- Modules that implement routing algorithms in a multiprocessor or network.

5.1.1.2 Organization of Network Simulation

A model network consists then of “nodes” connected by “links”. The nodes represent blocks, entities, modules, while the link represents channels, connections, etc. The structure of how fixed elements (i.e nodes) in a network are interconnected together is called topology.

Omnet++ uses a NED language, thus allowing for a more user friendly and accessible environment for creation and editing. It can be created with any text-processing tool (perl, awk, etc). It has a human-readable textual topology. It also uses the same format as the graphical editor, and supports sub module testing. Omnet++ allows for the creation of a driver entity to build a network at run-time by program.

Modular description of networks is given in NED language. The network description consists of a number of component descriptions such as channels, simple and compound module types. These component descriptions can be used in various network descriptions. Thus, it is possible to customize a personal library of network descriptions.

The files containing the network descriptions should end with a .ned suffix. The NEDC compiler translates the network descriptions into C++ code. Then, it is compiled by the C++ compiler and linked into executable simulation.

A NED description can contain the following components, in arbitrary number or order:

- Import statements;
- Channel definitions;
- Simple and compound module declarations;
- System module declarations.

Omnet++ follows a hierarchical module structure allowing for different levels of organization.

The topology of a node consists on two important aspects:

- OSI layers - The Data-Link, Network, Transport, Application
- Applications/protocols within a layer.

The physical layer is organized as follows:

Top-level network
Subnetwork (site)
LAN
node

5.1.1.3 User interfaces

The Omnet++ user interface is used with the simulation execution. Omnet++'s design allows the inside of a model to be seen by the user. It also allows the user to initiate and terminate simulations, as well as change variable inside simulation models. These features are handy during the development and debugging phase of modules in a project. The Graphical interface is a user friendly option in Omnet++ that allows access to the internal workings of the model.

The interaction of the user interface and the simulation kernel is through well defined interfaces. Without changing the simulation kernel, it is possible to implement several types of user interfaces. Also, without changing the model file, the simulation model can run under different interfaces. The user would test and debug the simulation with a powerful graphical user interface, and finally run it with a simple and fast user interface that supports batch execution.

The user interfaces are a form of interchangeable libraries. When linking into a created simulation executable, the user can choose the interface libraries it would like to use.

Currenly, two user interfaces are supported:

- Tkenv: Tk-based graphical, windowing user interface (X-Window, Win95, WinNT etc..);
- Cmdenv: command-line user interface for batch execution.

Simulation is tested and debugged under Tkenv, while the Cmdenv is used for actual simulation experiments since it supports batch execution.

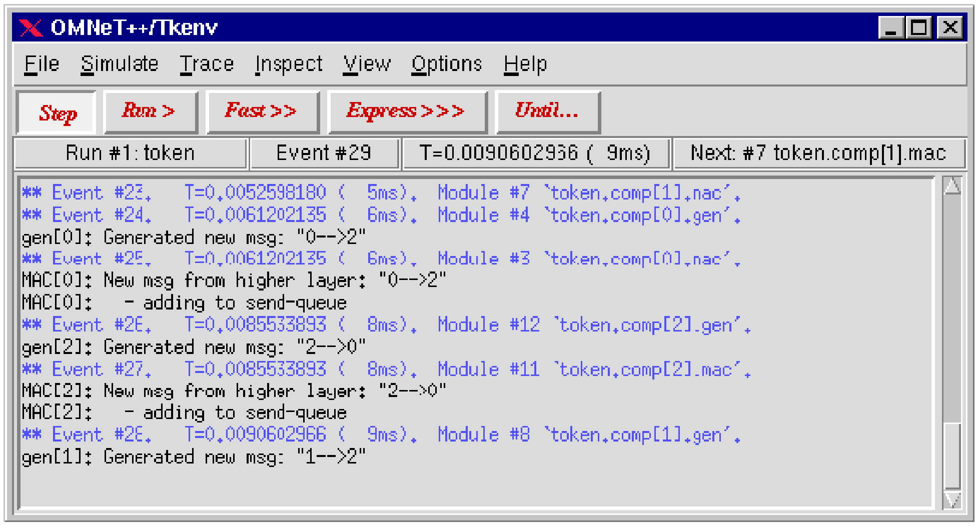
5.1.1.3.1 Tkenv

Tkenv is a portable graphical windowing user interface. Tracing, debugging, and simulation execution is supported by Tkenv. It has the ability to provide a detailed picture of the state of the simulation at any point during the execution. This feature makes Tkenv a good candidate in the development stage of a simulation or for presentations. A snapshot of a Tkenv interface is shown in Figure 59.

Important features in Tkenv are:

- Separate window for each module's text output;
- Scheduled messages can be watched in a window as simulation progresses;
- Event-by-event execution;
- Execution animation;
- Labeled breakpoints;
- Inspector windows to examine and alter objects and variables in the model;
- Graphical display of simulation results during execution. Results can be displayed as histograms or time-series diagrams;
- Simulation can be restarted;
- Snapshots (detailed report about the model: objects, variables etc.).

It is recommended for testing and debugging when used with gdb or xxgdb. Tkenv provides a good environment for experimenting with the model during executions and verification of the correct operation during the simulation program. This is possible since we are able to display simulation results during execution.



```
Run #1: token    Event #29    T=0.0090602936 (  9ms)    Next: #7 token.comp[1].mac
```

```
** Event #23.   T=0.0052598180 (  5ms).  Module #7  `token.comp[1].nac'.
** Event #24.   T=0.0061202135 (  6ms).  Module #4  `token.comp[0].gen'.
gen[0]: Generated new msg: "0-->2"
** Event #25.   T=0.0061202135 (  6ms).  Module #3  `token.comp[0].nac'.
MAC[0]: New msg from higher layer: "0-->2"
MAC[0]:   - adding to send-queue
** Event #26.   T=0.0085533893 (  8ms).  Module #12 `token.comp[2].gen'.
gen[2]: Generated new msg: "2-->0"
** Event #27.   T=0.0085533893 (  8ms).  Module #11 `token.comp[2].mac'.
MAC[2]: New msg from higher layer: "2-->0"
MAC[2]:   - adding to send-queue
** Event #28.   T=0.0090602966 (  9ms).  Module #8  `token.comp[1].gen'.
gen[1]: Generated new msg: "1-->2"
```

Figure 59 – Tkenv environment

5.1.1.3.2 Cmdenv

Cmdenv is designed primarily for batch execution. It is a portable and small command line interface that is fast. It compiles and runs on all platforms. Cmdenv simply executes all simulation runs that are described in the configuration file.

5.1.2 Simulations Performed

This section describes developments made inside the omnet simulator environment.

The first step taken was the definition of the network topology in a NED file. The topology defined contains terminals with mobility support to test the handovers. For this purpose, OMNeT++ supports two types of movement, with constant or variable speed: random way point and predefined path. The random way point moves the terminal in random direction, speed and time inside a virtual square, which delimitates the movement area. The predefined path allows the user to set the the exact movement the mobile terminal will have.

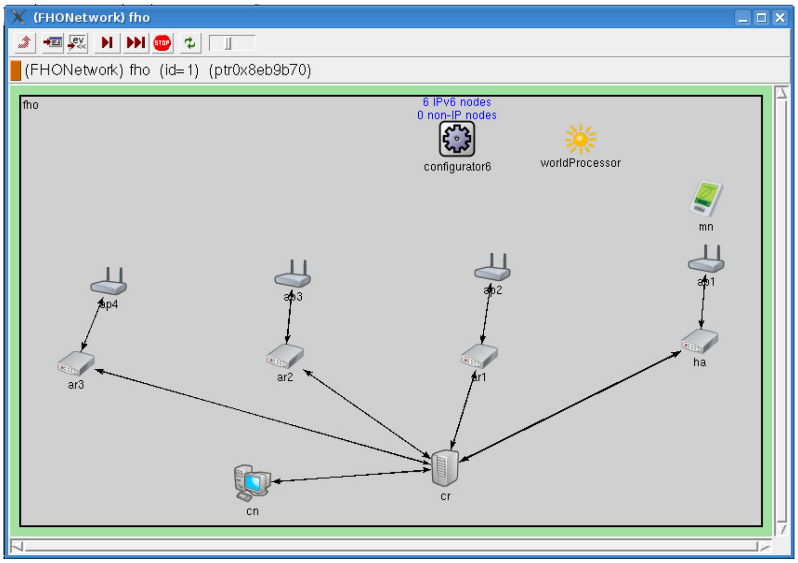


Figure 60 - Omnet developed scenario

Mobile IPv6 was integrated in the network depicted in Figure 60. When running this scenario, after some simulation time, the environment was crashing not allowing accomplishing a complete run. After some intensive debug sessions and mailing list support, the developers admitted that Mobile IPv6 and IPv6 stack were being re-developed due to integrity and stability problems. One of the main issues was that after several handovers, the Binding Update lifetime was reported to the home agent as 6 seconds which would make the binding expire very quickly; however, internally the timer was set correctly. Several attempts to correct these problems were done in the framework of this MsC Thesis and by the omnet++ developers without any success; the problem was mainly due to a non identified memory leak.

Since there was no schedule for the new development to be release and due to the instability of the simulator at this specific point, it was decided to postpone the simulation studies until these problems are corrected. At the present time, these problems are still not corrected by the development team, although there is some internal work to understand and fix these problems. Nevertheless, it was a great experience and a lot of knowledge was assimilated on the way simulators work and what they can provide to better understand the fundamentals and behaviours of the different architectures.

5.2 The Integrated Demonstrator

Thus, to test the implementation, perform conformance tests and evaluate the performance of the mobility architecture, a real testbed demonstrator was implemented. Our laboratory was responsible by the integration and test of the developed Daidalos software. During the lifetime of the Daidalos project, all software modules were developed and integrated into one demonstrator in our premises. This demonstrator includes the main entities needed for our architecture, the MT, ARs and QoS Broker.

The following subchapters describe the requirements and goals of this integration, its process, problems and recommendations for future integrations of large networks.

5.2.1 Integration Requirements and Goals

This section presents the integration process conducted to build the mobility subsystem and the problems faced during this process.

First, all the functional entities and software modules to develop were identified, and then the terminal mobility software was carefully analyzed. This analysis process allowed for the correct definition of the interfaces between the modules to be deployed. After this procedure, the development of the modules was distributed by different partners involved in the IST-Daidalos project.

The work here reported is centered in the MT's FHO Client, AR's FHO Attendant, ARM and Duplication and Merging, and in the QoS Broker modules.

Although modelling contributed for a well thought architecture and interface definition, this was not sufficient to avoid all integration problems.

After the modelling phase, the project entered a development phase. In this phase, an initial and very basic integration step was taken to verify that the software was being developed according with the specification, carefully looking at inputs, outputs and internal data procedures. At this point in time, mainly dummies were used to perform this first integration step.

When the development phase ended, a testing/integration phase started to check how the real modules interacted between them. The main objective of this phase was to check the correct working of each module and its functionalities according to the specification.

5.2.2 Integration Process

In order to efficiently run the integration process, we used the iterative approach depicted in Figure 61. We started by defining a set of tests and scenarios to which the software modules should respond in a strictly defined way. After this procedure, the method used consisted on testing a real software module interacting only with dummy software modules supplied by the developers. The next step was to incrementally start adding more functionality to the tested modules. In this process the behavior was always verified against the set of tests defined and checked if the system was fully compliant with what was specified within the activity.

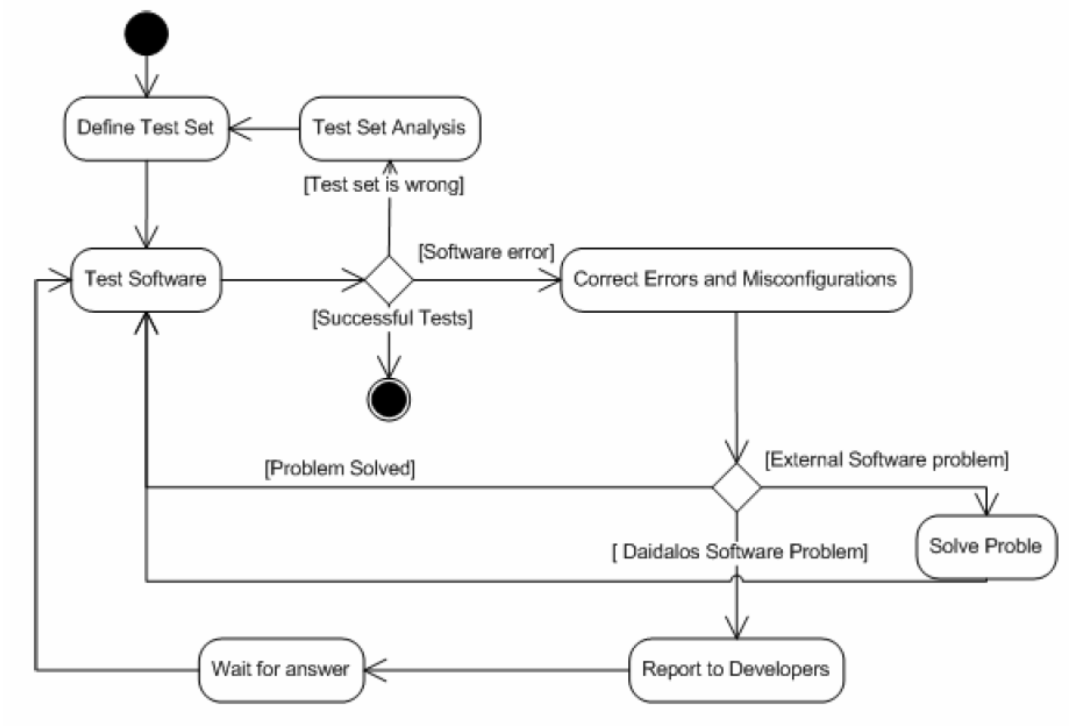


Figure 61 - Integration Process Flow Chart

5.2.3 Integration Problems

Integration is not usually an easy task and it is very time consuming. This section presents some of the concerns and problems faced during integration regarding system complexity and reliability, and the dependence on external software usually brings additional complexity. The developed demonstrator is not simple mainly due to its heterogeneous nature and software. In such a large integration effort, the development of efficient and simpler bootstrapping mechanisms is of large importance. Also, in real deployment, the developers need to have a very proactive role, all being part of the integration team.

5.2.3.1 System Complexity

One of the main problems with software integration was that it involves software from a large number of different with different knowledge, education and code style. This makes it very hard for the integration because it has to handle enormous software heterogeneousity. In an integration process that involves such a large amount of software modules, it is sometimes difficult to identify the cause of a malfunction.

To identify the cause of the problems, the use of dummy software provided the means to be able to control the inputs/outputs to the system and understand the problem. To ease the bootstrapping of all the modules to address the tests, several startup scripts were developed in order to simplify the procedure of starting, stopping and logging all the software modules.

5.2.3.2 System Reliability

Other main difficulty encountered during the integration effort was also to understand how robust was the modules' software. If the software modules were not working in a compliant way in some specific cases, we needed to replicate the environment variables that trigger this specific situation, gather logs from all the software modules and report to the software developers to correct the code. This procedure is not easy and is time consuming, since the response time to correct these situations is large. Also, replicating errors and gathering logs can also be a very time consuming task.

5.2.3.3 Dependences on external software

The IST Daidalos project has also several dependencies on external software. For instance, Mobile IP for Linux (MIPL) implementation is used in Daidalos with some modifications to support QoS and enhanced multi-homing. During the integration, we ran into some critical problems regarding Moblie IP behavior such as instability, problems with route optimization and tunnel deletion. Several approaches were tried to figure out the reason for this odd behavior, such as changing MIPL versions and changing Linux kernel version. We also tried different combinations of Daidalos software that could be influencing this malfunctioning of MIPL. In the end, we figure out that the problem was related to the Linux distribution the demonstrator computers were running at the time being. This distribution was Mandrake 10.0 Linux, which used an old *glibc* package with buggy threads implementation. The upgrade of this package was only possible migrating the whole demonstrator to Ubuntu Linux distribution, which had an adequate *glibc* package. This task turned out to be very complicated and time consuming to ensure compatibility with all the software, and to re-initialize all the integration process. After this migration, the main problems had been solved. As can be observed from this process, the cause of the problems is not always trivial to find; therefore, the integration team must be ready to identify, isolate and eliminate the problems.

5.2.4 Demonstrator

To validate the proposed architecture in terms of protocol design and basic functionality, we implemented and evaluated its performance in the test bed illustrated in Figure 62. The developed test bed contains 2 access networks, each one with wireless and wired (Ethernet) coverage, allowing for the support of intra- and inter-technology handovers.

The ARs (implemented in boxes with processor VIA C3 1 Ghz, 512 KB memory), provide wireless (2 of them in the access network) access by means of U.S. Robotics 11 Mbps PCMCIA Cards based on Prism2 Chipsets, and wired access through an Ethernet interface (1 interface each). The QoSBs, the central decision points, are supported by boxes with an Intel Pentium 4 3.4 Ghz and 1 GB memory capacity. The Home Agent acts as the Mobile IPv6 Home Agent entity for the MT and Correspondent nodes (Intel Celeron 2.66, 512 MB Memory). Each node (Correspondent and Mobile) is implemented with an IBM ThinkPad R52 equipped with U.S. Robotics 11 Mbps PCMCIA Cards based on Prism2 chipset and Ethernet interfaces.

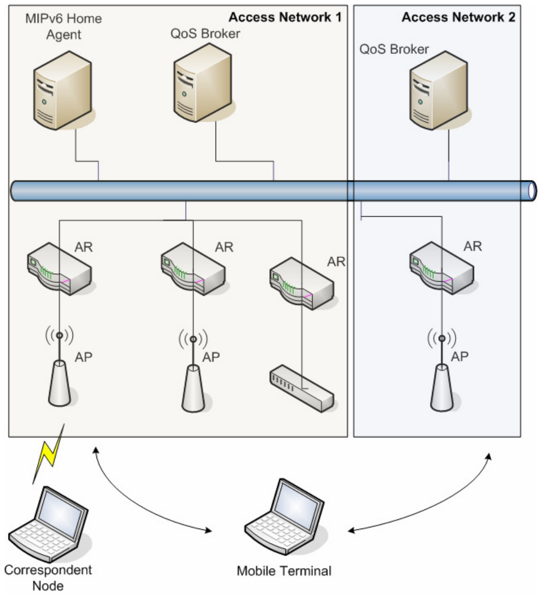


Figure 62 – Test bed Demonstrator

All the machines run GNU/Linux Ubuntu Breezy except one wireless access router which runs GNU/Linux Mandrake 10.1 Official. MIP version used was HUT v2.0.1. The correspondent node remains connected at a wireless AR and is not roaming, while the MT is capable of roaming freely between both wireless and wired ARs.

5.2.5 Tests Performed

According to the specification, the proposed architecture should provide fast and seamless mobility and be fully integrated in a QoS sub-system.

In order to evaluate the behaviour of the architecture, several tests were conducted. To accomplish these tests we considered handovers: i) between the same type of interface (wireless); ii) within the same access network and between different access networks; and iii) between WLAN and Ethernet access networks.

The tests performed include the evaluation of handover timings, and the influence of the handovers in data communications, both TCP and UDP based (TCP sequence numbers, losses, delay and jitter).

In order to measure all these parameters a set of tests were defined.

- Handover from WLAN to WLAN (intra techonology handover)
- Handover from WLAN to Ethernet (inter techonology handover)
- Fast Handover signaling
- Preparation and execution timing
- Impact on TCP streams
- Inter Access Network handovers

Each test was performed at least five times, and the presented values are the result of the mean value (in all relevant cases, also the 95% confidence interval is included).

The handover times presented in the following tests consist on the difference between the time the Fast Neighbour advertisement message is sent and the beginning of the handover procedure. The Stop Merging message is delayed for 3 seconds at the old AR in order to allow packets directed to the old care of address to be duplicated.

5.2.5.1 Signaling measurement without traffic – Intra-technology and intra-access network

The following signaling measurements give information on how the system behaves and how long it takes to perform some operations. This test was performed without any traffic flowing.

5.2.5.1.1 Mobile Terminal

Table 4 presents the messages exchanged by FHO Client, MTC and FHO Attendant. The values represent the instant time at which each event occurrs, such as receiving or sending a message, measured in seconds.

FHO_INITIATE	RtSolPr	PrRtAdv	FHO_HO_DEC	FHO_HO_REPORT	FBU	FHO_L2_DISC	FHO_L2_CONN	FNA	FHO_STOP_MERGING
0.000000	0.008524	0.035436	0.036187	0.045019	0.045625	0.046407	0.152335	0.187961	4.320178
0.000000	0.003802	0.029484	0.032867	0.045524	0.047211	0.057977	0.122532	0.147163	3.583238
0.000000	0.007954	0.026700	0.026718	0.035901	0.037873	0.039811	0.127761	0.167957	4.186103
0.000000	0.000900	0.027476	0.029282	0.041882	0.043568	0.055390	0.117595	0.175878	4.194816
0.000000	0.009706	0.029643	0.029841	0.053074	0.053242	0.053563	0.122224	0.178346	4.319115

Table 4 - Signaling Handover timing on Mobile terminal (seconds)

We observe that, from the mobile terminal side, the overall handover time takes 171,5 ms ± 15,3 ms. This table also shows that the L2 handover time, reported by RAL-WLAN (L2 Conn and L2 Disc) is 77,9 ms ± 18,7 ms. From the mobile terminal view, the authorization process (difference between the reception of Proxy for Router Advertisement and Router Solicitation for Proxy) takes 23,6 ms ± 3,9 ms. After the authorization, the mobile terminal needs to verify if the network is still available in order to perform the handover, which takes 13,3 ms ± 5,8 ms (FHO_HO_Report and FHO_HO_Decision).

Considering the Stop Merging message and taking out the 3 extra seconds, the complete Fast Handover signaling takes 1,12s ± 0,307s.

The time difference between Fast Neighbour Advertisement and the Stop Merging message is explained below.

5.2.5.1.2 Old Access Router

Table 5 presents the messages exchanged by the FHO Client, old AR FHO Attendant, old AR FHO@ARM and QoS Broker. The dark columns represent the messages at FHO Attendant and the light columns the messages at FHO@ARM.

RtSolPr	HO_Request	HO_Decision	PrRtAdv	FBU	HO_Report	HO_Status_Decision	Stop_Merging	Stop_Merging_MT
0.000000	0.001250	0.020790	0.023156	0.037136	0.038560	1.303470	1.305733	4.306602
0.000000	0.001210	0.021550	0.023683	0.043581	0.044930	0.573030	0.575177	3.576860
0.000000	0.001250	0.021200	0.023508	0.037928	0.039090	1.178730	1.181406	4.182981
0.000000	0.001260	0.021720	0.024115	0.042478	0.043640	1.187080	1.190150	4.191097
0.000000	0.001960	0.021760	0.024053	0.052616	0.054000	1.308760	1.311904	4.313513

Table 5 - Signaling Handover timing on Old Access Router (seconds)

The time that the QoS Broker takes to authorize the handover, between the HO_Request and HO_Decision message, is 20 ms ± 0,39 ms. The time between the reception of the FBU (from the terminal) and the HO_Status_Decision (from the QoSBroker meaning that the terminal is already at the new network) is 1,06 s ± 0,306 s.

The overall handover process at the old AR takes 1,114 s ± 0,306 s which, as expected, is very similar to the one in the MT (the difference is mainly due to the propagation times).

5.2.5.1.3 New Access Router

Table 6 presents the messages exchanged by FHO Client, new AR FHO Attendant, new AR FHO@ARM and QoS Broker. The dark column represent the messages at FHO Attendant and the light columns the messages at FHO@ARM.

COPS_Decision	FNA	COPS_Report
0.000000	1.132530	1.151142
0.000000	0.127830	0.147196
0.000000	1.121500	1.139721
0.000000	1.130790	1.149371
0.000000	1.128400	1.147119

Table 6 - Signaling Handover timing on New Access Router (seconds)

The overall process on the new access router takes 947 ms ± 447 ms, which is due to the uncertainty of the Fast Neighbour Advertisement messge. This message, due to kernel restrictions and link availability on the MT, takes aproximately 1 second to be sent. The reason for this to happen depends on several factors which are not controlled by the applications: kernel scheduling (link availability to the kernel - the message gets buffered and only when the link is completely available the message is sent).

5.2.5.2 Signalling measurement with traffic - Intra-technology and intra-access network

The following tests were performed with traffic flowing at a rate of 64 Kb/s – similar to voice calls. The traffic is used only to create a session and to understand how the QoS subsystem behaves in handovers with active sessions.

5.2.5.2.1 Mobile Terminal

Table 7 presents the messages exchanged by FHO Client, MTC and FHO Attendant.

FHO_INITIATE	RtSolPr	PrRtAdv	FHO_HO_DEC	FHO_HO_REPORT	FBU	FHO_L2_DISC	FHO_L2_CONN	FNA	FHO_STOP_MERGING
0.000000	0.000705	0.013650	0.015215	0.054759	0.054917	0.055334	0.222543	0.273658	4.735004
0.000000	0.004832	0.016228	0.016532	0.053283	0.053808	0.054091	0.198369	0.257435	4.241728
0.000000	0.002498	0.021241	0.021275	0.068094	0.068238	0.068622	0.232783	0.274003	4.285605
0.000000	0.012176	0.024851	0.025170	0.083133	0.083295	0.083708	0.261268	0.324003	4.347722
0.000000	0.003734	0.012626	0.012931	0.039068	0.043124	0.043388	0.213472	0.270490	4.263172
0.000000	0.000625	0.011104	0.011514	0.036022	0.036178	0.036589	0.214102	0.266032	4.275975

Table 7 - Signaling Handover timing with ongoing flows on Mobile terminal (seconds)

The overall and L2 handover times take more time than without traffic: with the traffic, the WLAN driver and the WLAN medium are busy, taking more time to attach to the new AP. The impact of traffic on the layer 2 handover time will be further analyzed.

In terms of the authorization time, it is below in this case than in the previous one. This shows that the QoS Broker is not a bottleneck, and that the extra processing is not noticeable.

Considering the Stop Merging message and taking out the extra 3 seconds, the complete Fast Handover signaling takes 1,35s ± 0,188s.

5.2.5.2.2 Old Access Router

Table 8 presents the messages exchanged by FHO Client, old AR FHO Attendant, old AR FHO@ARM and QoS Broker.

RtSolPr	HO_Request	HO_Decision	PrRtAdv	FBU	HO_Report	HO_Status_Decision	Stop_Merging	Stop_Merging_MT
0.000000	0.001700	0.005990	0.008368	0.053977	0.054920	1.727450	1.731324	4.732257
0.000000	0.001310	0.004470	0.006146	0.050105	0.051820	1.226580	1.229624	4.230608
0.000000	0.001220	0.005290	0.007609	0.065915	0.066670	1.277120	1.280369	4.281338
0.000000	0.001260	0.004990	0.006675	0.071705	0.072540	1.286990	1.290322	4.291828
0.000000	0.001200	0.005150	0.006699	0.039855	0.040920	1.252670	1.256222	4.257814
0.000000	0.001280	0.004480	0.006696	0.035784	0.036960	1.258240	1.261441	4.262967

Table 8 - Signaling Handover timing with ongoing flows on Old Access Router (seconds)

Again, the overall process at the old AR takes similar times to the ones at the MT.

5.2.5.2.3 New Access Router

Table 9 presents the messages exchanged by FHO Client, new AR FHO Attendant, new AR FHO@ARM and QoS Broker. The values are similar to the previous presented.

COPS_Decision	FNA	COPS_Report
0.000000	1.241850	1.717710
0.000000	1.199450	1.217595
0.000000	1.250210	1.267894
0.000000	1.259040	1.277630
0.000000	1.226350	1.243779
0.000000	1.230330	1.249470

Table 9 - Signaling Handover timing with ongoing flows on New Access Router (seconds)

5.2.5.3 Summary of the signaling measurements

This section presents an overview of the handover signalling times in all the presented scenarios. Table 10 shows the summary of the different handover timings in the different scenarios. All values are in seconds and represent the time elapsed since the beginning of the preparation phase for the different entities taking part on the handover process (please see Figure 28 for information on the handover phases). Each test was repeated at least 5 times and the results presented indicate the mean value of those tests. We should notice that the values obtained depend on the network conditions, such as wireless radio interferences and network load.

The tests were conducted in scenarios with and without traffic in the sessions. The traffic is composed by flows of 64 Kbps, similar to a voice call, to evaluate the influence of the load in the network.

		Intra-AN, Intra-tech No Traffic	Intra-AN, Intra-tech Traffic	Inter-AN, Intra-tech No Traffic	Inter-AN, Intra-tech Traffic	Intra-AN, Inter-tech No Traffic	Intra-AN, Inter-tech Traffic
MT	HO Authorization	23.571 ± 0.110	12.522 ± 0.087	15.165 ± 0.268	27.640 ± 0.044	14.362 ± 0.141	15.040 ± 0.265
	Link Availability	13.301 ± 0.164	38.620 ± 0.324	31.403 ± 0.150	26.623 ± 0.164	14.702 ± 1.978	16.936 ± 0.031
	L2 Handover	77.860 ± 0.524	166.801 ± 0.315	170.882 ± 0.254	178.21 ± 0.408	135.242 ± 1.993	122.273 ± 2.158
	Handover Execution	171.461 ± 0.556	277.604 ± 0.853	291.628 ± 0.520	322.856 ± 0.655	199.425 ± 2.428	248.124 ± 1.449
	Complete HO Time	1120.690 ± 307.326	1358.201 ± 4.813	1501.689 ± 12.763	1327.153 ± 0.500	626.345 ± 17.101	864.156 ± 14.184
oAR	HO Authorization	20.018 ± 0.011	3.733 ± 0.012	17.192 ± 0.032	22.599 ± 0.044	4.266 ± 0.006	3.810 ± 0.009
	HO Execution	1067.466 ± 8.597	1285.285 ± 4.886	1429.170 ± 12.749	1239.169 ± 0.537	581.308 ± 17.078	815.215 ± 14.401
	Complete HO Time	1114.211 ± 8.599	1342.802 ± 4.914	1495.700 ± 12.806	1314.010 ± 0.455	619.525 ± 17.101	851.840 ± 14.414
nAR	Complete HO Time	946.910 ± 12.537	1329.013 ± 4.904	1473.789 ± 12.792	1287.652 ± 0.0466	604.984 ± 17.127	838.014 ± 14.396

Table 10 - Handover timing (miliseconds)

5.2.5.3.1 Mobile Terminal

In an intra-AN scenario, the FHO signalling process in the MT takes 172 ms, from which 80 ms are used to perform the L2 handover. This means that actually the FHO signalling takes 92 ms. This is the total time, which consists on the MT requesting an handover, which has to be analyzed and approved by the QoS Broker, getting the decision, informing the network that it is going to move, handover at Layer 2, and inform the new network about its presence there.

The time required for handover authorization is then approximately 13 to 28 msec. The check of the link availability consists on accessing and scanning the wireless medium ensuring that the desired AP is still available. This time is usually more than 20 msec for intra-technology scenarios. L2 handover time consists on the handover between the old and the new AP. This time represents the difference between the request and the report from the driver. This value is in the order of one hundred of msec and it imposes a bottleneck on our signalling protocol, since it cannot overcome L2 issues. The handover execution period represents the time elapsed from the *Router Solicitation for Proxy* message until the first message sent on the new link informing the new AR of its presence. This value is in the order of 250 msec and includes all the above periods. Finally, the complete handover time is usually larger than 1 sec for intra-technology scenarios. This is mostly due to the MIPv6 recovery period.

When comparing the timing values with and without traffic in the network, we observe that, as expected, generally the values are smaller in a network without traffic. However, there are some specific cases where the opposite happens, since the traffic is not large enough to adversely affect the performance of the handover signalling.

Comparing the intra-AN with the inter-AN scenarios, we observe that the handover process takes more time in inter-AN scenarios, since it contains communication between QoS Brokers, which slows down the process. However, the difference between the scenarios is not very large.

The difference between the intra- and inter-technology scenarios is noticeable mainly in terms of link availability and L2 handover time. To perform the check of link availability, it is required to access and scan the wireless medium; on the inter-technology scenario, this scanning is not always necessary. When moving from WLAN to Ethernet, interface scanning is not needed, saving time that is reflected in the results presented. In terms of L2 handover, the results obtained are usually lower for inter-technology scenarios, since the architecture only needs to activate the new interface which might be already activated, reducing the time consumed at this point.

Finally, in terms of the handover execution period, we would expect it not to be dependent on the scenarios. Due to concurrency on the software stack, these results have a little variation among scenarios.

5.2.5.3.2 Old Access Router

For the measures in the old AR, the handover authorization is the period between the request and the decision involving the QoS Broker, and communication between QoS Brokers in the inter-AN scenario. In this case the traffic is not expected to influence the results. The small differences are due to specific wireless network characteristics, such as radio interferences. The authorization time in inter-AN scenarios is again slightly larger due to the communication between the QoS Brokers.

The handover execution phase is noticeably lower for intra-AN scenarios when compared to inter-AN. Moreover, inter-technology scenarios also decrease this time, due to the smaller times to perform local link availability and L2 handover.

5.2.5.3.3 New Access Router

For the measures in the new AR, the time measured is the time between the notification of a handover by the QoSB and the notification by the MT informing about its presence on the new link. Again, the same trend is observed when compared to the handover execution time in the nAR.

5.3 Influence in Data Communications

In this section the influence of the proposed architecture on traffic flows will be analyzed. For this purpose TCP and UDP traffic was used while performing handovers.

5.3.1 TCP traffic

To analyze the behaviour of the architecture regarding TCP streams, we established a TCP stream with 256 Kb/s rate. With the communication active, we performed intra- and inter-technology handovers.

As can be observed in Figure 63, which shows the TCP sequence numbers in inter-technology handovers, the sequence number curve has a constant slope during handovers, which means that TCP arrival rate is constant during handovers, and there is no interruption or degradation of the flow. This can only be achieved because both interfaces are up and running. Although Mobile IPv6 is only controlling one of the interfaces at each instant, both interfaces are able to send and receive packets causing no degradation. As soon as the binding update gets to the correspondent nodes, they update their neighbour cache and start sending packets to the new interface.

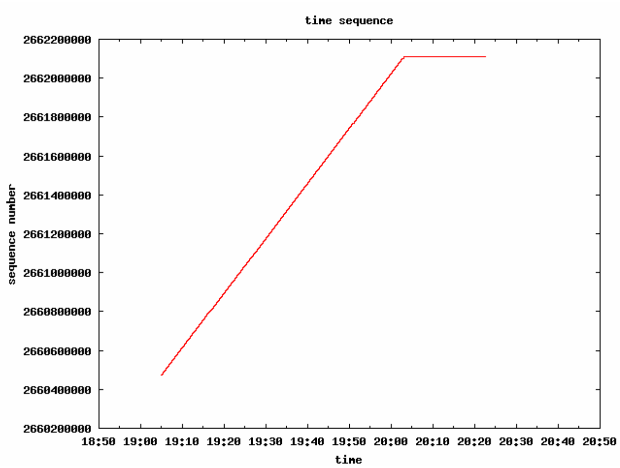


Figure 63 - Inter-Technology TCP Sequence Numbers vs Time

This scenario provides a fast and seamless mobility with integrated QoS resource allocation.

On the other hand, Figure 64 shows the situation when the handover occurs within the same technology and the same interface. Here there are some interruptions which reflect the handover period. In this case, there is no connectivity during some time. After that period, TCP starts the normal retransmission of lost segments. The interruption is due to MIPv6 recovery and L2 handover period, when there are no packets flowing to the MT.

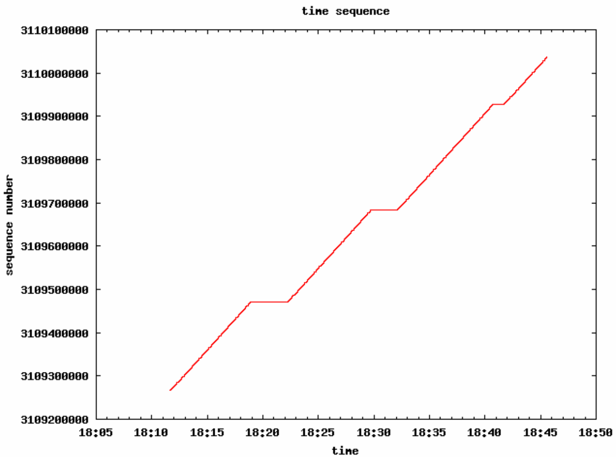


Figure 64 - Intra-Technology TCP Sequence Numbers vs Time

5.3.2 UDP Traffic

This section studies the impact of the mobility protocol in UDP flows at different rates. UDP traffic tends to represents a portion of applications which provide real time traffic to users, which has the most significant impact on degradation awareness.

5.3.2.1 Intra-Technology Analysis

We used the tool MGEN [mgen] to generate the UDP traffic. MGEN was configured to send a periodic flow of 8, 16, 32 or 64 packets per seconds each of 512 bytes size, totaling 32 Kbps, 64 Kbps, 128 Kbps and 256 Kbps. Larger bandwidths were not used, since there was no stability on the association concerning the wireless cards (Prism 2 chipset) whenever a handover occurred. The card often is not able to associate to the new AP whenever there is a constant bandwidth usage of more than 256 Kbps.

To evaluate the measurements, three kinds of scenarios were tested. First, only Mobile IPv6 was running and the handover was performed by issuing a *iwconfig* command to change AP. Afterwards, a test with the Daidalos scenario was performed but no duplication of packet at the old AR was done. Last, we performed the second test with duplication.

5.3.2.1.1 Mobile IPv6 recovery time

This section covers the time Mobile IPv6 takes to recover from a handover regarding packet loss, jitter and delays for different traffic speeds.

Considering UDP flows of 32 Kbps (Figure 65 to Figure 67), the overall handover time takes approximately 3.76 seconds, resulting on an average of 61.43% of packet loss within that period.

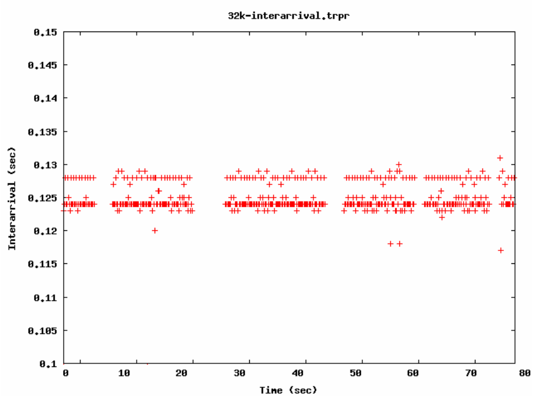


Figure 65 - Standalone MIPv6: Packet Inter arrival at 32Kbps

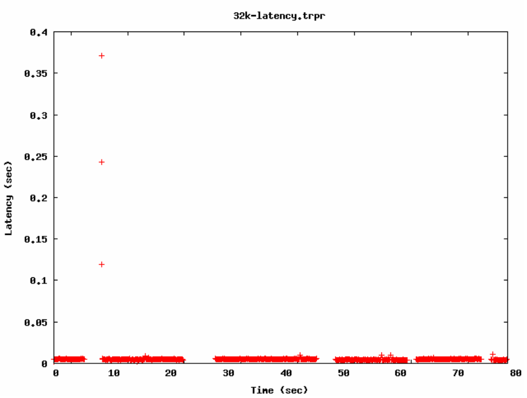


Figure 66 - Standalone MIPv6: Packet Latency at 32Kbps

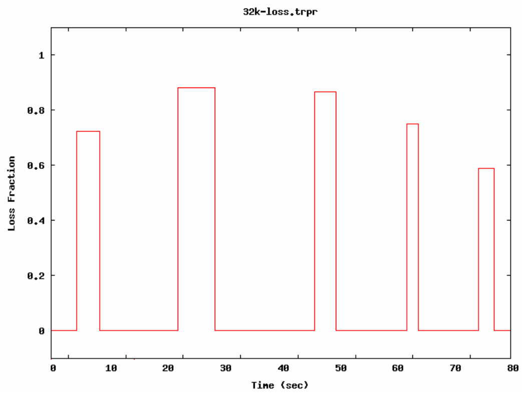


Figure 67 - Standalone MIPv6: Packet Loss at 32Kbps

With UDP flows of 64 Kbps, each handover at 64Kbps takes approximately 3.29 seconds resulting on an average of 69.02% of packet loss. These values are similar to the previous with UDP flows of 32 Kbps.

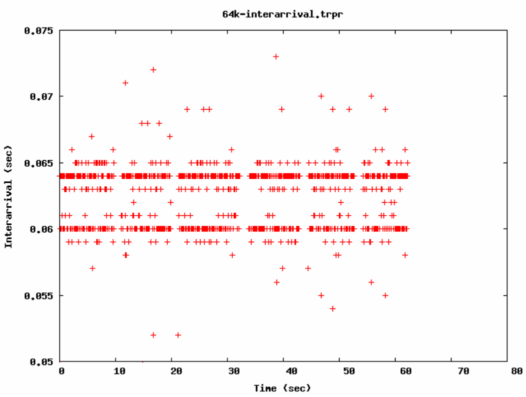


Figure 68 - Standalone MIPv6: Packet Inter arrival at 64Kbps

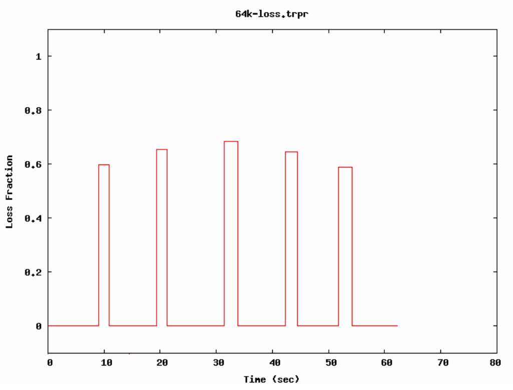


Figure 69 - Standalone MIPv6: Packet Loss at 64Kbps

Finally, with UDP flows of 128 Kbps, each handover takes approximately 2.3 seconds resulting on an average of 73.03% of packet loss, and with 256 Kbps, it takes approximately 3.13 seconds resulting on an average of 78.77% of packet loss.

From these graphics and values we observe that, for every handover there is an interruption of the traffic flowing, and it depends on several factors such as link availability, interface availability, router advertisement arrival and so on, which causes Mobile IPv6 to delay its recovery. The loss increases with the rate; this is due to the amount of traffic in the "air" which slows down the association and packet delivery, causing the driver to back-off more often. The handover time is not adversely afected with the increase in the UDP traffic rate.

5.3.2.1.2 Terminal Mobility Software without duplication

This section covers the time the mobility protocol without duplication takes to recover from a handover. Figures Figure 70 to Figure 75 depict the arrival rate, delay and losses for the UDP flows of 32 Kbps and 256 Kbps. Table 11 includes the mean handover time and losses with different rates of UDP flows.

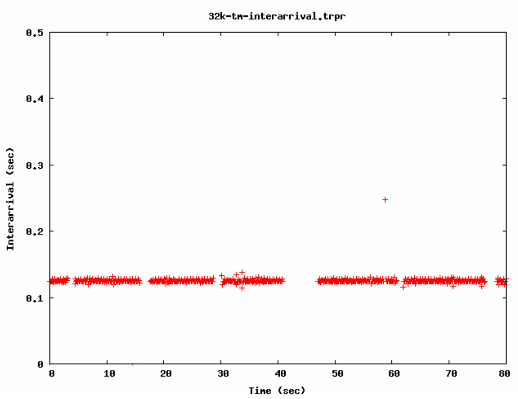


Figure 70 - Terminal Mobility without duplication: Packet Inter arrival at 32 Kbps

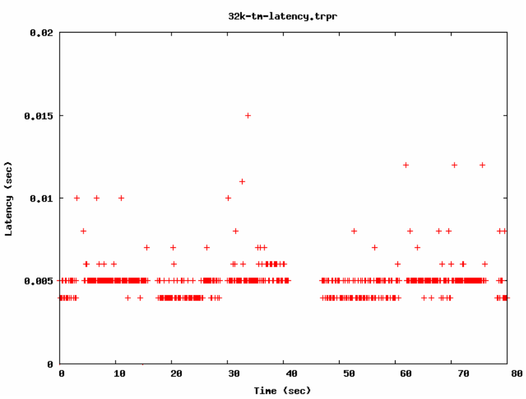


Figure 71 - Terminal Mobility without duplication: Packet delay at 32 Kbps

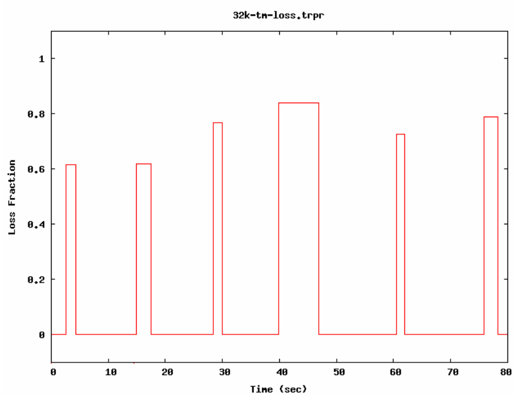


Figure 72 - Terminal Mobility without duplication: Packet Loss at 32 Kbps

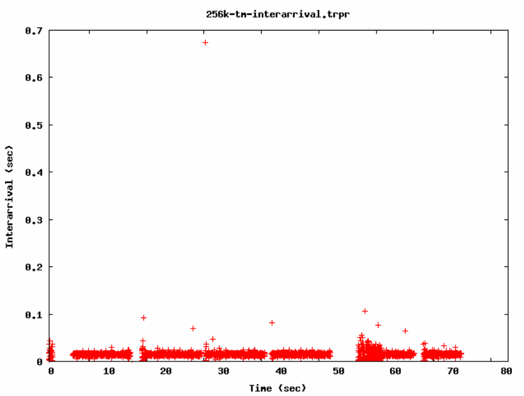


Figure 73 - Terminal Mobility without duplication: Packet Inter arrival at 256 Kbps

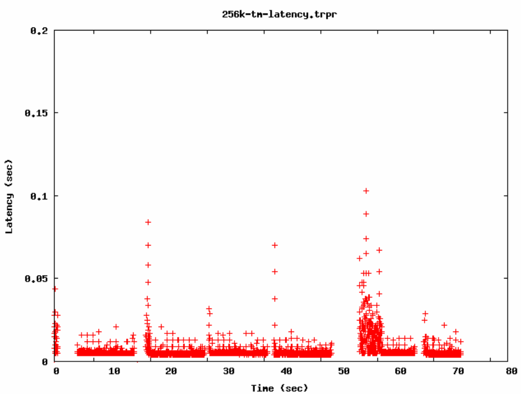


Figure 74 - Terminal Mobility without duplication: Packet delay at 256 Kbps

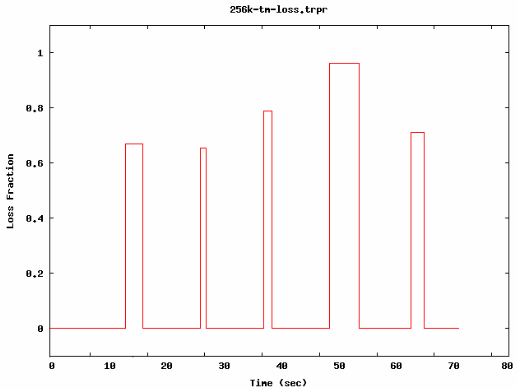


Figure 75 - Terminal Mobility without duplication: Packet Loss at 256 Kbps

Rate (Kbps)	Time (s)	Packet Loss (%)
32	2.00	69.53
64	1.98	73.93
128	1.94	62.06
256	1.92	70.70

Table 11 - Measurements for the Terminal Mobility without duplication scenario

The measurements taken in this scenario shows that the handover time is mostly the same. This is due to some optimizations made in Mobile IPv6 code so that it can send the binding update as soon as possible. This translates to pre-configuration of the routes and IPv6 addresses while performing the Layer 2 handover. The use of these enhancements provided a more stable Mobile IPv6 version, as is obvious by the similar values of the handover times and the packets lost at different rates.

5.3.2.1.3 Terminal Mobility Software with duplication

This section covers the time Terminal Mobility Software with duplication takes to recover from a handover regarding packet loss, jitter and delays for different traffic speeds. The measurements obtained are presented on Table 12.

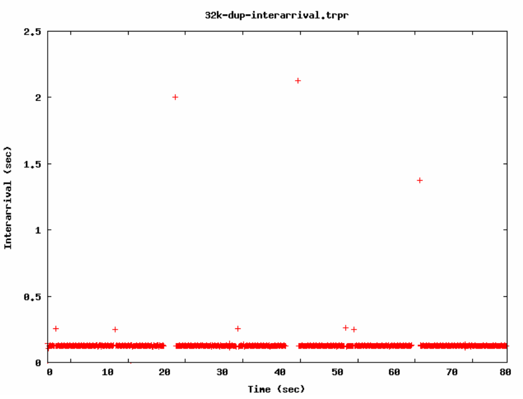


Figure 76 - Terminal Mobility with duplication: Packet Inter arrival at 32 Kbps

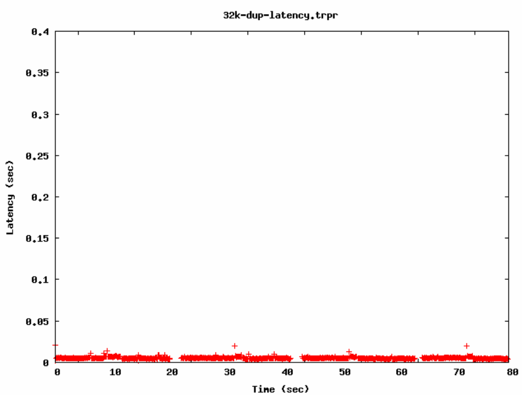


Figure 77 - Terminal Mobility with duplication: Packet delay at 32 Kbps

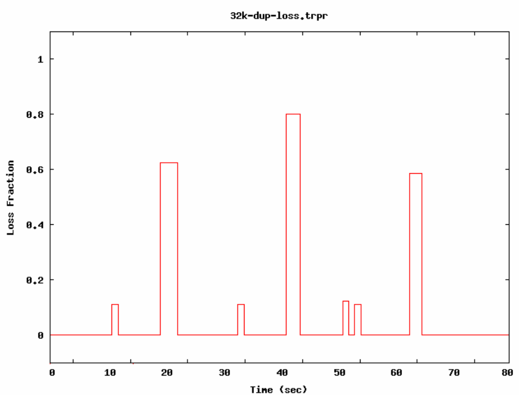


Figure 78 - Terminal Mobility with duplication: Packet Loss at 32 Kbps

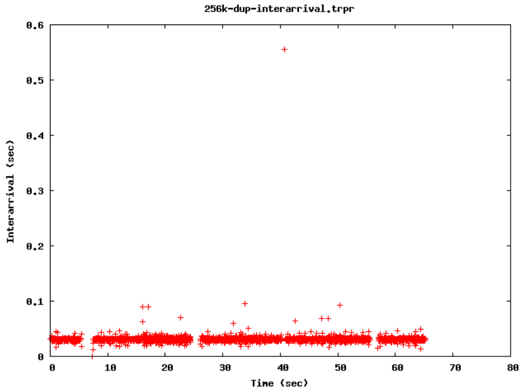


Figure 79 - Terminal Mobility with duplication: Packet Inter arrival at 256 Kbps

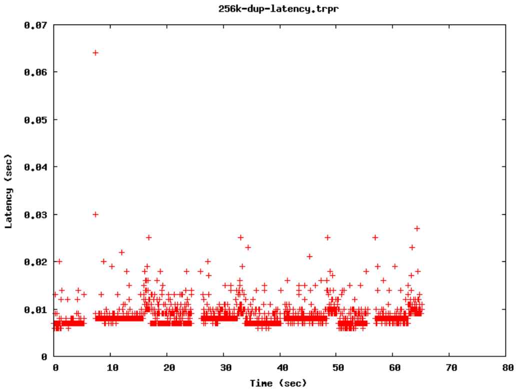


Figure 80 - Terminal Mobility with duplication: Packet delay at 256 Kbps

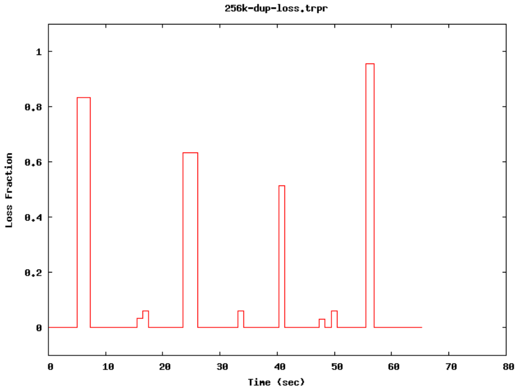


Figure 81 - Terminal Mobility with duplication: Packet Loss at 256 Kbps

Rate (Kbps)	Time (s)	Packet Loss (%)
32	1.08	11.45
64	1.08	5.96
128	1.22	6.78
256	1.25	6.33

Table 12 - Measurements for the Terminal Mobility with duplication scenario

The values presented refer only to the handovers where duplication occurred.

These graphics show the importance of duplication in this architecture. These times were taken in a scenario where the terminal was handing over from one AR to another, but only one AR had the duplication enabled. The graphics are really impressive on the effectiveness of duplication in such scenarios. The little spikes show an handover with duplication and the large ones show handover with no duplication available. This represents an improvement about 90% on the packet loss.

Table 13 summarizes the average values and the confidence interval at 95% of handover time and packets lost obtained for the different mobility protocols, considering different bit rates of the flows. The table also shows the improvement (in %) obtained with the terminal mobility (TM) sub-system, without (TM) and with duplication of packets (TM+Dup). The results are once more the average of at least five test runs. In each run one handover was performed around each 10 seconds period.

	MIPv6		TM		Improvement over MIPv6		TM+Dup		Improvement over MIPv6	
	Time (s)	Packets Lost	Time (s)	Packets Lost	Time %	Loss %	Time (s)	Packets Lost	Time %	Loss %
32 Kbps	3.76 ± 0.05	18.48 ± 0.92	2.00 ± 0.01	11.12 ± 0.11	46.81	39.83	1.08 ± 0.01	0.98 ± 0.01	46.25	83.53
64 Kbps	3.29 ± 0.07	36.29 ± 2.54	1.98 ± 0.01	23.42 ± 0.23	39.75	35.46	1.08 ± 0.01	1.03 ± 0.01	45.45	91.94
128 Kbps	2.30 ± 0.01	53.75 ± 0.53	1.94 ± 0.01	38.53 ± 0.39	15.65	28.32	1.22 ± 0.01	2.65 ± 0.26	37.11	89.08
256 Kbps	3.13 ± 0.05	157.98 ± 7.90	1.92 ± 0.03	86.88 ± 2.61	38.72	45.01	1.25 ± 0.01	5.06 ± 0.05	34.90	91.05

Table 13 - UDP Datagrams: Handover time and loss at handover execution

The handover period measured relates to the time that terminal is moving from one AP to another, which has no optimization whatsoever, thus resulting in similar conditions to Mobile IPv6 stand alone test. The differences in the handover time shown in the figures presented above demonstrate how a high dynamic environment combined with Mobile IPv6 instability can influence test conditions reducing the ability to get the proper results, leading to huge handover times, and thus resulting on large amount of packet loss.

Harmonizing the handover period by taking out the clearly identified failures of Mobile IPv6, the handover period is somehow smaller than the raw Mobile IPv6 scenarios. At 32 Kbps, it takes around 2.0 seconds to regain connectivity resulting on 11.12 packets lost; and at 256 Kbps it takes around 1.92 seconds to regain connectivity resulting on 86.88 packets lost. This improvement is due to some minor optimizations performed by the FHO client module, such as configuration of the new CoA at the interface and the new gateway before the handover.

5.3.2.2 Inter-Technology Analysis

Table 14 represents the number of packets lost per run on UDP traffic at 256 Kbps.

	Run											
	1	2	3	4	5	6	7	8	9	10	11	Mean
Lost Ping Replies	0	0	0	0	0	0	0	0	0	0	0	0

Table 14 - Inter Technology packet loss analysis

As expected, since both interfaces are available during the handover execution phase, there is no packet loss (there is no need to duplicate packets since they do not get lost in the way to the old CoA). This scenario is the most suitable one since no packets are lost and Mobile IPv6 is not time constrained to recover and rebind the HoA to the new CoA; the traffic flow does not suffer from the handover, and there is no extra delay or jitter noticeable at the packet flow.

5.4 Architecture and Implementation Limitations

This section presents some limitations due to both architecture and implementation aspects. Most of these limitations represent problems that appeared during testing and integration which, obviously, were not taken into consideration during the design phase. Also, there are problems that were known during design phase, but that revealed very difficult or even impossible to cope in the implementation phase.

Figure 82 identifies the different stages of the handover process. This figure shows the timing required by MIPv6 and by our mobility sub-system approximately to scale.

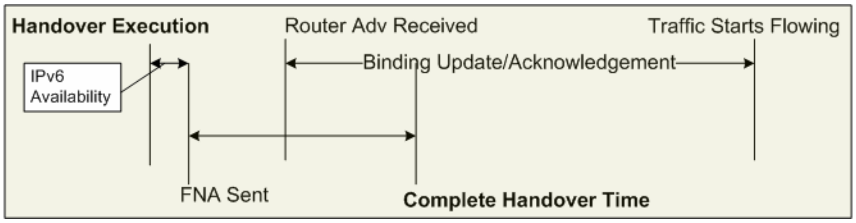


Figure 82 - Handover timings

In a fully optimized system (and not a simple running prototype), the timing identified as *FNA Sent* would be done in parallel with the MIP-timing *Binding Update/ Acknowledgment*, and would internally trigger the MIP state machine, forcing traffic to start flowing in the MT much sooner. In fact, from the moment the *Binding Update* is sent, the system would be able to process traffic. As it is implemented now, the MIP state machine requires the external *Binding Acknowledgement* message for allowing traffic to flow, since our system is only overlaid to the MIP software, and not integrated with it. Further enhancements for minimizing the IPv6 stack availability could thus be performed. Although the improvements in timing are significant, note that the FNA timing would always be dependent on local paths, while the *Binding Acknowledgement* will depend on the total path between the MT and the Home Agent. Thus, in a real network, with optimized code, the potential improvements could be even better, and no fundamental impairment to seamless handovers seems to be present, if the MIP stack integration is performed.

Note that these same arguments can be made even if other mobility protocol was being used (such as HIP). That was the main reason behind the non-MIP dependence of our code. Our mobility mechanism, if integrated, would improve most global mobility protocols. In fact, the MIP integration was not finally performed due to identified stability deficiencies in the MIP software release used, and this integration is now being done on a more recent release of MIP. In those optimized situations, the handover time will be limited by the physical layer handover, added with the couple of dozen milliseconds required for the control signalling. Note that, if soft handover is possible at the physical layer, our system allows for the temporal overlap of most of these phases.

Some added improvement could nevertheless still be achieved in the mobility scheme, even without a close linkage with the underlying mobility protocol. This improvement would be easily achievable if scanning time (see section 3.4.1.1) is removed by introducing added information in CARD (such as 802.11 channels). In order to scan for available ARs/APs and retrieve from the network the surrounding candidates, CARD needs to be connected to the network at power up. This is somehow limiting because it overpasses security and access control, but for a prototype it is adequate.

5.5 Summary and conclusions

This chapter presented the work carried out regarding the integration and evaluation of the mobility protocol presented in Chapter 3.

In terms of the integration effort, most of the problems existent concerned the integration of foreigner software. Several tests were performed to discover the software errors and identify the responsible modules. Although much effort was addressed to the solution of software problems, many problems were not related to the Daidalos software. For example, the MIPL versions required several updates to achieve a stable testbed.

In terms of the evaluation, the developed testbed allowed to perform intensive tests and gather performance results in different scenarios, with intra- and inter-access network scenarios, intra- and inter-technology, and with different traffic types (UDP with different rates and TCP). The results are somehow encouraging and show that it is possible to have this solution in the future, where full integration between mobility and QoS is supported. We faced some constrains, mainly due to the lack of complete integration with the global mobility protocol and technology dependent delays at layer 2 handovers. These constraints show that there is still some margin to enhance the architecture to make it as seamless as possible regarding intra-technology handover. Inter-technology handovers proved to be an excelent example that can benefit from this architecture; in this case, the handovers are totally seamless and the integration with the QoS subsystem provides the roaming user to handover without any degradation or extra delay.

Chapter 6 Conclusions

This master thesis had as main objective to specify, evaluate and implement IPv6 Fast-Mobility architectures with the goal to provide transparent mechanisms of fast mobility.

In the first stage of this work several mobility protocols were studied and analysed in order to better understand the associated requirements. By the similarities of Fast Handovers for Mobile IPv6, we decided to use it as a basis and extend it to provide integrated fast-mobility with a quality of service subsystem.

The proposed architecture is able to provide fast and seamless handovers to heterogeneous next generation operators while maintaining the quality of service in use by the user. It supports mobile terminal initiated handovers and network initiated handover both coordinated by a central entity which interfaces with the QoS and A4C subsystem. The MIHO is executed upon detection of a new connection point, which best suits the needs of the current user regarding price, bandwidth and technology while the NIHO is ordered due to network congestion which requires a network optimization enforced by the redistribution of the attached mobile terminals among the access points reached by each one of them. The incorporation of mechanisms such as Intelligent Interface Selection, Candidate Access Router Discovery, Duplicate and Merging, and Performance Management makes it suitable for heterogeneous networks with distributed decision points.

The extension of the resulting protocol was submitted as a draft to the IRTF working group [melia05] considering the network initiated handover scenario. This IRTF draft has already expired, but some of its ideas have been exploited in the mipshop group.

After this analysis, the protocol was extended to two scenarios with QoS integration, one using multicast to transport the traffic to the neighbours access routers and other using the same multicast transport but no QoS assured. The second proposed architecture is entirely supported by the previous architecure although the method for duplicating the traffic among the neighbour routers recurs to a multicast network. The third architecture complementing the two previous ones was specified, resulting in a scenario for high mobility users with no QoS warranties nor authorization but allowing users to move wherever they desire in the neighbourhood.

This master thesis further detailed a mobility architecture developed inside the Daidalos project. The architecture had a good performance, even without optimum integration with MIPv6, demonstrating a great improvement compared to a standalone solution. Furthermore, this provides integration with QoS, and thus may comply with the requirements that subscribers will want to have in the future, no degradation and the contracted quality of service across several technologies. The subscriber can roam freely in the heterogeneous environment without noticing it, and supporting the "always best connected" paradigm.

This architecture also aims to be independent of the global mobility protocol which is in use on the network, since it doesn't require integration with any mobility protocol, although for optimization purposes a small integration with the mobility protocol currently in use is advised in order to trigger the location update. This operation minimizes the time that the mobility protocol realizes that it is connected to a new network and issues the location updates by itself.

The evaluation presented in this master thesis was constrained by the Mobile IPv6 behaviour, since the sub-system developed was purposely made independent on the underlying mobility protocol. This led to apparently long handover times – while in reality the mobility process is quite fast. The amount of packet loss experienced was due to Layer 2 and Layer 3 availability at handover execution phase, where no optimization was performed. Optimized linkage with Mobile IPv6 can easily reduce this problem. Note that, in the inter-technology scenario no optimization is needed.

Furthermore, this shows that both network and mobile terminal initiated handovers can be supported with minimal cost in terms of handover process.

6.1 Future Work

The Daidalos Project will have its second phase which has already started. In this second phase we plan to extend the proposed protocol in order to have a close integration with IEEE 802.21 standards [802.21]; this will provide media independency and a compliant protocol in the future. Better integration with security mechanisms, virtual identities and multihoming are also envisioned as future work.

References

[3gpp04] 3GPP. Network architecture. In “3GPP TS 23.002 V6.4.0”, June 2004.

[802.11] IEEE 802.11 Working Group
URL: http://www.ieee802.org/11/

[802.21] IEEE 802.21 Working Group
URL: http://www.ieee802.org/21/

[aguiar06] R. L. Aguiar, S. Sargento, et al, “Scalable QoS-Aware Mobility for Future Mobile Operators”, IEEE Communications Magazine, June 2006.

[ambient] FP6 IST Ambient Consortium
URL: http://www.ambient-networks.org

[banerjee05] Banerjee, N, et al, “SIP-Based Mobility Architecture for Next Generation Wireless Networks”, 3rd International Conference on Pervasive Computing and Communications, 2005.

[banerjee06] Banerjee, N, et al, “Seamless SIP-Based Mobility for Multimedia Applications”, IEEE Network, March/April 2006.

[bless04] R. Bless, J. Hillebrand, C. Prehofer, and M. Zitterbart, “Quality-of-service signaling for next generation ip-based mobile networks”. volume 42 of Communications Magazine, pages 72–79. IEEE, June 2004.

[campbell02] A. Campbell, J. Gomez, S. Kim, and C. Wan., “Comparison of IP micro-mobility protocols.” In IEEE Wireless Communications, vol 9, pages 72–82, February 2002.

[daidalos] FP6 IST Daidalos Consortium
URL: www.ist-daidalos.org

[dh] Diffie-Hellman Algorithm
URL: http://www.rsa.com/rsalabs/node.asp?id=2248

[dutta04] Dutta, A et al, “Fast-Handoff Schemes for Application Layer Mobility Management”, 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2004.

[ethereal] Ethereal: A Network Protocol Analyzer
URL: http://www.ethereal.com

[everest] FP6 IST Everest Consortium

URL: http://www.everest-ist.upc.es

[garcia06] C. Garcia, A. Cuevas, et al, "QoS Support on fourth generation networks", IEEE Latin American Transactions, March 2006.

[gomes04] Diogo Gomes, et al, "A transsignaling strategy for QoS support in heterogeneous networks", ICT'2004, 11th International Conference on Telecommunications, Fortaleza, Brazil, ISBN: 3-540-22571-4, pp 1114-1121, Springer Verlang.

[hillebrand04] J. Hillebrand, C. Prehofer, et al, "Quality-of-Service Signaling for Next-Generation IP-Based Mobile Networks", IEEE Communications Magazine, June 2004.

[jaehnert05] Juergen Jaehnert et all, "Moby Dick: A Pure-IP 4G Architecture", Computer Communications, Elsevier Computer Communications, Vol 28/9 pp 1014-1027, Jun 2005.

[laganier06] J. Laganier, L. Eggert, "Host Identity Protocol (HIP) Rendezvous Extension", draft-ietf-hip-rvs-05.txt, June 2006..

[leu03] S. Leu and R. Chang. "Integrated service mobile internet: Rsvp over mobile ipv4&6". In Mobile Networks and Applications, Volume 8,Issue 6, pages 635–642, December 2003.

[lo04] S. Lo, G. Lee, W. Chen, and J. Liu. "Architecture for mobility and qos support in all-ip wireless networks". In IEEE Journal on Selected Areas in Communications, vol. 22, no. 4, May 2004.

[marques03] Victor Marques et al. "An ip-based qos architecture for 4g operator scenarios". IEEE Wireless Communications, June 2003.

[marques05] Victor Marques et al, "Evaluation of a Mobile IPv6-based Architecture Supporting User Mobility, Quality of Service and AAAC in Heterogeneous Networks", IEEE Journal on Selected Areas in Communications, , Vol. 23, no. 11, Nov 2005.

[melia04] T. Melia, R. Schmitz, and T. Bohert. "Tcp and udp performance measurements in presence of fast handovers in an ipv6-based mobility environment". In 19th World Telecommunications Congress (12-15 September 2004, Seoul, Korea), September 2004.

[melia05] T. Melia, R. Aguiar, N. Sénica, "Network initiated handover in fast mobile ipv6 handovers", draft-melia-mobopts-niho-fmip-01.txt, IETF draft, July 2005.

[mgen] MGEN: The Multi-Generator Toolset
URL: http://pf.itd.nrl.navy.mil/mgen

[mobiwan] MobiWan : NS2 extensions to study mobility in Wide-Area IPv6 networks
URL: http://www.inrialpes.fr/planete/mobiwan

[nikander04] P. Nikander, J. Arkko, T. Henderson, "End-Host Mobility and Multi-Homing with Host Identity Protocol", draft-nikander-hip-mm-02, July 2004.

[ns2] NS2 : The Network Simulator

URL: http://www.isi.edu/nsnam/ns

[nursimloo05] Nursimloo D.S. et al, “Integrating fast mobile Ipv6 and SIP in 4G networks for real-time mobility”, 13th IEEE International Conference on Networks 2005.

[omnet] Omnet++, András Vargas.
URL: www.omnetpp.org

[opnet] OPNET
URL: http://www.opnet.com

[pana06] D. Forsberg, et al, “Protocol for Carrying Authentication for Network Access (PANA)”, draft-ietf-pana-pana-13.txt, IETF Internet Draft, December 2006.

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

[RFC2212] S. Shenker, C. Partridge, R. Guerin , “Specification of Guaranteed Quality of Service”, RFC 2212, September 1997.

[RFC2215] S. Shenker,J. Wroclawski, “General Characterization Parameters for Integrated Service Network Elements”, RFC 2215, September 1997.

[RFC2460] S. Deering, R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification”, RFC 2460, December 1998.

[RFC2463] A. Conta, S. Deering, “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification”, RFC 2463, December 1998.

[RFC2474] K. Nichols, S. Blake, F. Baker, D. Black, ”Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers”, RFC 2474, December 1998.

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

[RFC2748] D. Duhram, J. Boyle, R. Cohen, S. Herzog, R. Rajan, A. Sastry, “The COPS (Common Open Policy Service) Protocol”, RFC 2748, January 2000.

[RFC3261] J. Rosenberg, et al, “SIP : Session Initiation Protocol”, RFC 3261, June 2002.

[RFC3376] B. Cain, et al, “Internet Group Management Protocol, Version 3” RFC 3376, October 2002.

[RFC3775] D. Johnson, C. Perkins, and J. Arkko, ”Mobility support in IPv6”, RFC 3775, IETF, June 2004.

[RFC3776] J. Arkko, et al, ”Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents”, RFC 3776, June 2004.

[RFC3971] J. Arkko, J. Kempf, et al, “SEcure Neighbor Discovery SEND”, RFC 3971, IETF Internet Draft, July 2004.

[RFC3973] A. Adams, et al, “Protocol Independent Multicast – Dense Mode (PIM-DM) : Protocol Specification (Revised)”, RFC 3973, January 2005.

[RFC4066] M. Liebsch et al, “Candidate Access Router Discovery (CARD)”, RFC4066, July 2005.

[RFC4068] R. Koodli (ed), et al, ”Fast Handover for Mobile IPv6”, RFC 4068, IETF, July 2005.

[RFC4140] Hesham Soliman et al., “Hierarchical mobile IPv6 mobility management (hmipv6)”, RFC 4140, August 2005.

[RFC4423] R. Moskowitz, P. Nikander, “Host Identity Protocol (HIP) Architecture”, RFC 4423, May 2006.

[RFC4429] M. Moore, “Optimistic Duplicate Address Detection (DAD) for IPv6”, RFC 4429, April 2006.

[RFC4601] B. Fenner, et al, “Protocol Independent Multicast – Sparse Mode (PIM-SM) : Protocol Specification (Revised)”, RFC 4601, August 2006.

[senica05] Nuno Sénica, Justino Santos, Susana Sargento, Rui L. Aguiar. “QoS-Aware Fast Handover Optimization Supported by Multicast Networks”. Revista do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro. Setembro de 2005.

[senica06] Nuno Sénica, Justino Santos, Susana Sargento, Rui L. Aguiar. “Mobility Between Heterogeneous Networks: Integration”, CSMU 2006 – Conferência sobre Sistemas Móveis e Ubíquos, Guimarães, Junho 2006.

[senica07] Nuno Sénica, Justino Santos, Susana Sargento, Rui L. Aguiar. “Mobility Between Heterogeneous Networks: Integration and Evaluation”, Special Issue on Mobile and Ubiquitous Computing of The Mediterranean Journal of Computers and Networks, January 2007.

[shelby00] Zach D. Shelby, Dionisios Gatzounas, Andrew Campbell, ”Cellular IPv6”, IETF Draft, draft-shelby-seamoby-cellularipv6-00, November 2000.

[umts] 3GPP Document. “25301.700 Radio interface protocol architecture version 7”. April 2006.

[vieth] M. Vieth, O. Menzel, K. Jonas, I. Miloucheva, J.F. Placido, H. Santos, E. Guainella, “Learning for efficient handover of QoS based mobile multicast services”. CITSA 2005: Int. Conf. on Cybernetics and Information Technologies, Systems and Applications Orlando, July 2005.

[yasukawa01] S. Yasukawa, J. Nishikido, and K. Hisashi. “Scalable mobility and qos support mechanism for ipv6-based real-time wireless internet traffic”. In Globecom, volume 6, 2001.