Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2011

Carlos Eduardo Magalhães Guimarães

Descoberta de Serviços Independentes do Acesso para Redes Heterogéneas

Discovery of Media Independent Services for Heterogeneous Networks

Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2011

Carlos Eduardo Magalhães Guimarães

Descoberta de Serviços Independentes do Acesso para Redes Heterogéneas

Discovery of Media Independent Services for Heterogeneous Networks

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requesitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Prof. Doutor Rui Luís Andrade Aguiar, Professor do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro, e do Prof. Doutor Pedro Alexandre de Sousa Gonçalves, Professor da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro

o júri / the jury

presidente / president	**Prof. Doutor José Luís Guimarães Oliveira** Professor Associado da Universidade de Aveiro
vogais / examiners committee	**Prof. Doutora Marília Pascoal Curado** Professora Auxiliar do Departamento de Engenharia Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra
	Prof. Doutor Rui Luís Andrade Aguiar Professor Associado com Agregação da Universidade de Aveiro (orientador)
	Prof. Doutor Pedro Alexandre de Sousa Gonçalves Professor Adjunto da Universidade de Aveiro (co-orientador)

agradecimentos / acknowledgements

Agradeço à minha família, pelo apoio incondicional.

Agradeço aos meus amigos pela paciência para me aturar.

Agradeço a todos os meus professores da Universidade de Aveiro, que contribuiram para o meu crescimento pessoal e académico, em particular ao Prof. Rui Aguiar e ao Prof. Pedro Gonçalves.

Agradeço o apoio e colaboração do Daniel Corujo e de todos os membros do ATNOG.

Palavras Chave Descoberta; IEEE 802.21; Pontos de Serviço; Mobilidade

Resumo

A recente proliferação de nós móveis com múltiplas interfaces sem fios e a constituição de ambientes heterogéneos possibilitaram a criação de cenários complexos onde os operadores de rede necessitam de disponibilizar conectividade para diferentes tipos de redes de acesso. Assim, a norma IEEE 802.21 foi especificada de forma a facilitar e optimizar os procedimentos de *handover* entre diferentes tecnologias de acesso sem perda de conectividade. Para cumprir o seu propósito, a norma disponibiliza serviços chamados *Media Independent Handover* e que permitem o controlo e a obtenção de informação de diferentes ligações. A configuração estática destes serviços por parte do nó móvel torna-se ineficiente devido aos múltiplos cenários possíveis. Desta forma, o nó móvel deve descobrir nós da rede que providenciem serviços de mobilidade e as suas capacidade de uma forma dinâmica. Nesta dissertação, um conjunto de mecanismos para descoberta de serviços de *handover* independentes do acesso são analisados, implementados e avaliados em termos de duração e quantidade de informação trocada. Um novo mecanismo de descoberta de entidades locais é também proposto e avaliado, demonstrando que a sua utilização aumenta o desempenho e requer a troca de menos quantidade de informação.

Keywords Discovery; IEEE 802.21; Point of Service; Mobility

Abstract The recent proliferation of mobile nodes with multiple wireless interfaces, in addition to the creation of heterogeneous environments, created complex scenarios where network operators need to provide connectivity for different kinds of access networks. Therefore, the IEEE 802.21 standard has been specified to facilitate and optimize handover procedures between different access technologies in a seamless way. To fulfil its purpose, it provides Media Independent Handover services which allow the control and gathering of information from different links. The static configuration of these services by the MN becomes inefficient due to the amount of possible scenarios. Thus, the MN must discover the network-supporting nodes and their capabilities in a dynamic way. In this work, a series of proposed Media Independent Handover discovery procedures are analyzed, implemented and evaluated in terms of duration and amount of exchanged information. In addition, a novel discovery procedure for local entities is proposed and evaluated, showing that its deployment increases the performance and requires less information exchanged.

Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Description
1G	1st Generation
2G	2nd Generation
3G	3rd Generation
3GPP	3rd Generation Partnership Project
AAA	Authentication, Authorization and Accounting
ABC	Always Best Connected
AID	Action Identifier
ANSDF	Access Network Service Discovery Function
AP	Access Point
API	Application programming interface
AR	Access Router
ARP	Address Resolution Protocol
BSC	Base Station Controller
BSS	Basic Service Set
BTS	Base Transceiver Station
CAR	Candidate Access Router
CARD	Candidate Access Router Discovery
CDMA	Code Division Multiple Access 2000
CoA	Care-of Address
DHAAD	Dynamic Home Agent Address Discovery
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol version 6
DNS	Domain Name System
DS	Distributed System
EDGE	Enhanced Data Rates for Global Evolution
ESS	Extended Service Set
FA	Foreign Agent
FMIPv6	Fast Mobile IPv6
FQDN	Fully Qualified Domain Name
GMSC	Gateway Mobile Switching Centre
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HA	Home Agent
HMIPv6	Hierarchical Mobile IPv6
ICMP	Internet Control Message Protocol

Continues on the next page

Acronym	Description
ID	Identifier
IE	Information Element
IP	Internet Protocol
IS	Information Server
L2	Layer 2
L3	Layer 3
LAN	Local Area Network
LCoA	Local Care-of Address
LLC	Logical Link Control
LMA	Localized Mobility Anchor
LMD	Local Mobility Domain
LOC	Locator
MAC	Medium Access Control
MAG	Mobile Access Gateway
MAP	Mobility Anchor Point
MDE	Mobility Decision Engine
MICS	Media Independent Command Service
MID	Message Identifier
MIES	Media Independent Event Service
MIH	Media Independent Handover
MIHF	Media Independent Handover Function
MIHO	Mobile-initiated Handover
MIIS	Media Independent Information Service
MIP	Mobile IP
MIPv4	Mobile IPv4
MIPv6	Mobile IPv6
MN	Mobile Node
MoS	Mobility services
MSC	Mobile Switching Centre
NAI	Network Access Identifier
NAPTR	Naming Authority Pointer
NIC	Network Interface Card
NIHO	Network-initiated Handover
Opcode	Operation Code
OS	Operating System
PBA	Proxy Binding Acknowledge
PBU	Proxy Binding Update
PHY	Physical layer
PMIPv6	Proxy Mobile IPv6
PoA	Point of Attachment
PoS	Point of Service
PRL	Parameter Request List
QoS	Quality of Service
RADIUS	Remote Authentication Dial In User Service
RCoA	Remote Care-of Address
RDF	Resource Description Framework
SAP	Service Access Point

Continues on the next page

Acronym	Description
SCTP	Stream Control Transmission Protocol
SID	Service Identifier
SPARQL	SPARQL Protocol and RDF Query Language
SSID	Service Set Identifier
STA	Station
TCP	Transmission Control Protocol
TID	Transaction Identifier
TLV	Type-length-value
UDP	User Datagram Protocol
UIR	Unauthenticated information request
UMTS	Universal Mobile Telecommunications System
WLAN	Wireless Local Area Network

Chapter 1

Introduction

The proliferation of wireless and cellular access networks creates a new paradigm in communications, allowing mobility across coverage areas of different kinds of access networks. The heterogeneous environment created with the increased opportunities provided by all these access technologies for connecting to the Internet, stimulated the development of mobile nodes (MNs) with multimode connectivity capabilities (i.e., devices supporting multiple technologies).

This led to the exploitation of different online connection opportunities, motivated by the deployment of distinct technology access networks provided by different operators. For example, users can be provided with a high transmission rate through Wireless Local Area networks (WLANs) indoors (such as home or public facilities), while cellular networks provide a wider coverage area, making the user always connected outdoors. In the same geographical area, wireless networks of different technologies and cellular networks may coexist, allowing the mobile users to discover and connect to different link technologies, offering different connectivity opportunities to access services and run applications. For example, a mobile user that wishes to make a voice call will probably be connected to a cellular network, since it provides a voice service. However, assuming that during the voice call it wishes to send a video, the connection can be established through a WLAN network in order to send the data, since it can provide high-speed data connections. Such environments allow the maximum use of the respective features and capabilities of both networks, providing not only an ubiquitous but also an Always Best Connected (ABC) [1] experience to the user.

Parallel to this, and taking as example the previous case, it is expected that the voice call is not interrupted for sending the video. In the same way it is expected that the mobile users are able to access online content while on the move, motivating the need for transparent session continuity mechanisms while crossing different points of attachment (PoAs) to the network.

Since the Internet Protocol (IP) was not developed to support mobility scenarios (due to addressing and routing issues) several protocols have been developed at layer 3 (L3) of the network stack, such as Mobile IPv6 (MIPv6) [2] and Proxy Mobile IP (PMIPv6) [3]. These aim to support mobility and to allow increased connectivity scenarios. However, the complexity of managing different link connections on the MN, as well as the challenges to the network operators' ability to provide optimum connectivity, are not overcome by the proposed IP mobility protocols. To overcome these issues the IEEE 802.21 was specified, whose main purpose is to facilitate handover management in heterogeneous environments. To achieve its

purpose, it provides a framework that allows the abstraction of the specificities of each link technology. Making use of that abstraction, information from those links can be retrieved to higher-layer decision entities, allowing them to control the links. The coupling of IP mobility protocols with IEEE 802.21 enabled the optimization and the exploitation of several mobility scenarios.

Environments created by mobility are characterized by being dynamic and unpredictable, i.e., it creates many and different possible scenarios, beyond the imaginable. Static configurations are very limited in this aspect and may not cover all possible cases and, therefore, the specification of mechanisms for discovering the entities, which are responsible for managing mobility and Media Independent Handover (MIH) services, is required. For example, this has special importance when the MN connects to a new network for the first time and has no prior knowledge of available network controlling entities therein. This is where this work focuses, providing a study and evaluation of MIH discovery mechanisms, and proposing a novel local discovery procedure for automatically detecting link interfaces in MIH-enabled entities.

1.1 Objectives and Methodology

The main goal of this work is the study, elaboration and implementation of discovery mechanisms for MIH entities in a given domain, using layer 2 (L2) or L3 solutions over an 802.21 open-source implementation (named ODTONE[1]).

Initially, the developed work relies on the study of several aspects related to the mobility research area:

- identification of the enabling technologies for mobility;
- identification of the existing mobility protocols;
- identification of the available discovery mechanisms.

Parallel to this, the IEEE 802.21 standard is studied, highlighting the aspects related to the discovery procedures. Therefore, the discovery solutions for L2 and L3 of the network stack are identified.

The identified mechanisms are then implemented over the ODTONE framework, which is able to run over Linux, Windows and/or Android operating systems.

Finally, the implementation is tested and evaluated over several scenarios, in terms of performance and duration of each mechanism. The obtained results are compared against each other in order to identify the advantages and disadvantages of each one.

1.2 Contributions

The work developed under this dissertation was subject of one accepted article, named "Using an open-source IEEE 802.21 implementation for network-based localized mobility management" [4], which was published on the IEEE Communications Magazine (September 2011), as well as a paper entitled "Evaluation of Discovery Mechanisms for Media Independent

[1]Open Dot Twenty One (ODTONE) - http://atnog.av.it.pt/odtone

Handover Services", that is pending acceptance for the 2nd IEEE Workshop on Convergence among Heterogeneous Wireless Systems in Future Internet (CONWIRE 2012).

This work also contributes to the implementation of several discovery modules, which were integrated with an open source IEEE 802.21 implementation, the ODTONE.

1.3 Outline

The dissertation is organized in 7 chapters. Here is a brief description of what this document contains:

- **Chapter 2:** contains a brief description of two wireless technologies that can be used in heterogeneous handovers scenarios. An overview of IP mobility aspects is also made, emphasizing the research on discovery mechanisms.
- **Chapter 3:** provides a description of the IEEE 802.21 standard, highlighting the aspects that are more closely related to this work.
- **Chapter 4:** describes the MIH discovery mechanisms based on L2 and L3 approaches. A novel mechanism to discover local MIH entities is also presented.
- **Chapter 5:** explains the implementation details of the MIH discovery mechanisms.
- **Chapter 6:** presents the evaluation of the implemented discovery mechanisms. It also evaluates the impact of the ODTONE in the discovery process.
- **Chapter 7:** describes the key point of the developed work, highlighting its main contributions and points out further work.

Chapter 2

Enabling Technologies for Mobility

In the recent years, several advances have been made [5] allowing the proliferation of different kinds of access networks (including wireless access) and the development of new multimedia applications (such as voice and video over IP (Skype), video sharing (Youtube) and television over IP (Tivo)). In addition, with the development of wireless and cellular networks, a new paradigm in communications became available for users, allowing them to be mobile and to remain constantly connected. Merging cellular networks with the Internet enabled the creation of a heterogeneous environment, which stimulated the development of devices supporting multiple technologies. Since each technology has its limitations (for example, in terms of throughput, coverage area or even quality of service), this heterogeneous environment provides a scenario where the multi-technology devices can select the most appropriate technology at each time, providing the best quality of service for the user.

However, it implies the development of mobility management solutions that are not dependent on specific link technologies (i.e., solutions that enable uniform control of to all link technologies). Based on this assumption, it is clear that L2 solutions are limited to the technology and, therefore a L3 solution is also needed, more specifically a solution based on the IP protocol. In this way, several L3 mobility protocols have been proposed in the standard bodies such as, MIP [6] or PMIPv6 [3]. This has been the subject of extensive work and, therefore, several studies have been made on how to facilitate and to optimize mobility procedures. However, despite that these L3 mobility solutions operate at the IP level, they still require interactions with the link layer, in order to evaluate and execute full fledged mobility scenarios. In order to facilitate and optimize handovers in heterogeneous environments, the IEEE 802.21 standard [7] has been proposed, providing an abstract access to the different link layers.

This chapter provides a brief overview of two frequently used wireless technologies, involved in mobile procedures for local and cellular connectivity. Next, the main mobility management protocols will be described, as well as the discovery mechanisms that support them.

2.1 Wireless Technologies

In this section, a brief overview of two of the most used wireless technologies is presented: the IEEE 802.11 and 3rd Generation (3G) networks. Although the 3G technology is not directly involved in this work, it is presented here as an alternative to the IEEE 802.11 and

to highlight the importance of an abstract interface to each access technology, since each one has particular aspects.

2.1.1 IEEE 802.11

The IEEE has defined a wireless standard, named IEEE 802.11 [8] (or Wi-Fi), whose purpose is to provide wireless connectivity to fixed, portable and mobile stations within a local area network (LAN). To achieve its purpose, it specifies a medium access control (MAC) and several physical layers (PHY), as well as the protocols needed to support a wireless network within a LAN.

A WLAN (Figure 2.1) can be seen as a cellular network where each cell consists of a Basic Service Set (BSS), that corresponds to the coverage area of its Access Point (AP) (also called as Base Station) in which member stations (STA) are able to communicate with each other. Therefore, the AP is the entity on IEEE 802.11 networks that represents the PoA. Within the BSS a STA can change its location transparently to the upper layers and without affecting its communications. However, if a STA moves out of the BSS, it can no longer directly communicate with other STAs of the BSS. Although each BSS can be independent of the other, they can be part of an extended network that can be built with multiple BSSs. This is achieved by the Distributed System (DS), which is a backbone that interconnects the APs of each BSS. The union of multiple BSSs by a DS allows the creation of a wireless network with arbitrary size and complexity, named Extended Service Set (ESS). To the Logical Link Control (LLC), the ESS network is seen as a single BSS and therefore, the STA can move from one BSS to another (within the ESS) without losing its ability to communicate with another STA that belongs to the same ESS.

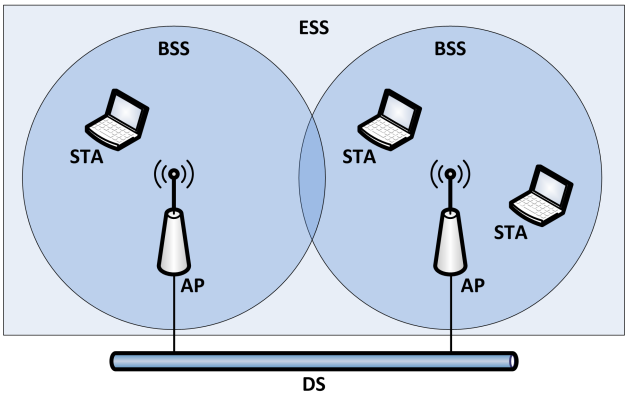

Figure 2.1: WLAN architecture

WLANs that deploy APs as the central entity of the wireless network are often called infrastructured WLANs. If the STAs group themselves in a wireless network it is usually referred as ad-hoc network. In this work, only aspects related to the infrastructure WLAN will be mentioned, although they can also be related with ad-hoc networks.

The IEEE 802.11 defines a mechanism that enables STAs to find available network APs. It is called wireless scanning and provides the following methods:

- **Passive Scanning:** the STA listens to each channel during a certain period of time, seeking for Beacon Frames from any AP. These frames are sent periodically from an AP

and enables the STAs to discover new APs in the range and to synchronize with the AP.

- **Active Scanning:** this method involves the active participation of the STA in the discovery process, i.e., the discovery process is initiated by the STA. It sends a Probe Request Frame with the a broadcast Service Set Identifier (SSID) or a specific SSID. The AP, after receiving these messages, verifies if the SSID is set to its own SSID or to a broadcast one. If the Probe Request meets these criteria, the AP replies with a Probe Response directly to the STA that generated the discovery process, enabling it to discover the AP.

Figure 2.2 depicts the steps for a STA to access an existing BSS. Initially, it needs to discover and synchronize with the AP, using one of the wireless scanning methods described above. Based on the discovered APs, the STA decides and selects to which AP it wants to connect and then authenticates itself with the AP. After authenticating with the AP, the STA sends an association request frame to the AP, which responds with an association response frame. Finally, the STA is associated with the AP (i.e., the STA has been established a L2 connection and it can initiate the procedures to join the subnet to which the AP belongs).

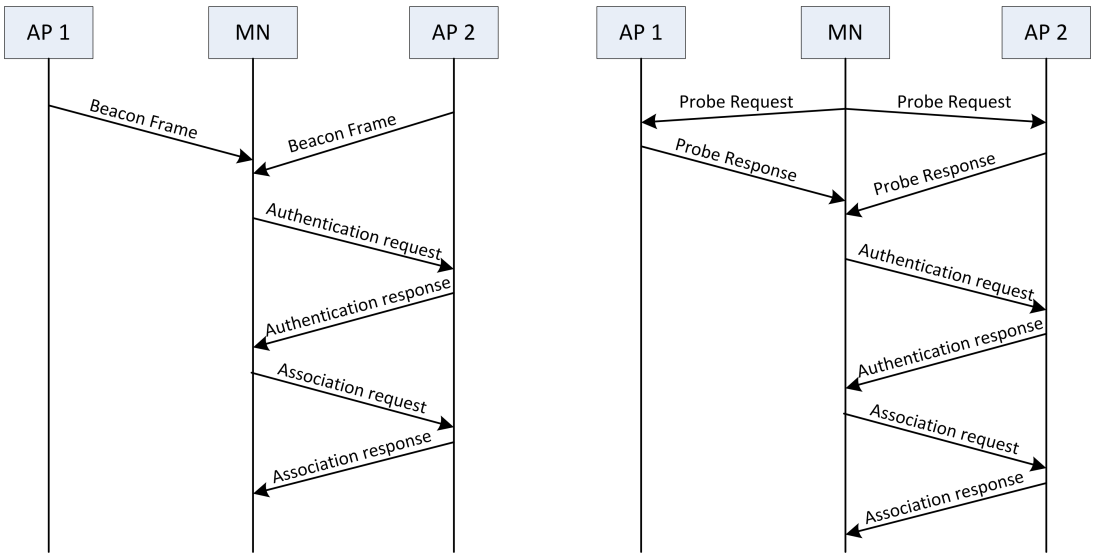

Figure 2.2: IEEE 802.11 Association Process

2.1.2 3G Systems

Cellular networks [5] were designed primarily for voice-only traffic, in the first and second generations (1G and 2G) systems. However, these have been later extended (2.5G) to support data traffic. The currently deployed systems correspond to third generation (3G) systems, which support voice and data communications, but compared with 2.5G systems have a greater data capability and higher speed radio access links.

The architecture of cellular networks (Figure 2.3) consists in several geographic coverage areas, known as cells, each one containing a Base Transceiver Station (BTS) that is responsible for sending and receiving signals from the MN within the cell. Then, several BTSs are

connected to a Base Station Controller (BSC), although it can be physically located with the BTS. The BSC function is to allocate radio channels to the MN, perform paging and the handover of MNs, and therefore it represents the PoA on cellular networks. The BSC and correspondent BTS constitute a Global System for Mobile Communications (GSM) base station system (BSS). The BSCs are then grouped in by the Mobile Switching Centre (MSC). This network entity is responsible for the user authentication and accounting, handover and call establishment and tear down. The MSCs are connected to a gateway, named Gateway MSC (GMSC), that plays a central role in connecting the cellular network to the larger public telephone network. This represents the core of cellular networks for voice communications.

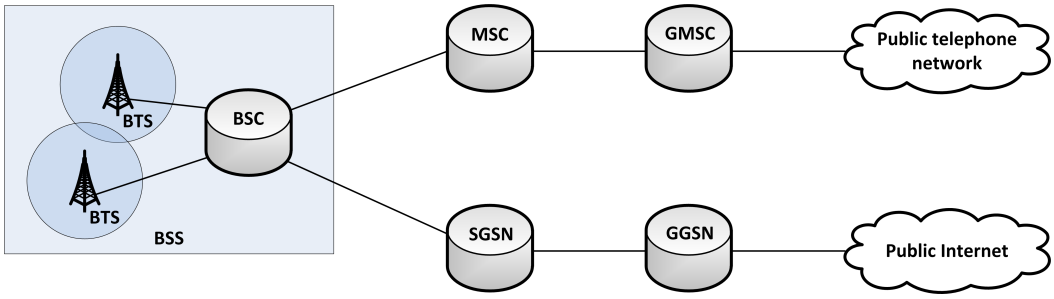

Figure 2.3: Basic cellular network architecture

In order to provide Internet access via cellular networks, two new network entities were added, that are responsible to connect the cellular network to the Internet: the serving General Packet Radio Service (GPRS) support node and the gateway GPRS support node. Its operation is similar to the MSC and GMSC, respectively.

At radio access networks for data in 2G systems, GPRS was introduced and to increase the data rates capabilities the Enhanced Data Rates for Global Evolution (EDGE) was introduced. Actually, for 3G networks, the Universal Mobile Telecommunications Service (UMTS) and the Code Division Multiple Access 2000 (CDMA-2000) are two major standards for 3G systems.

2.2 Mobility

When a MN changes the PoA with which it is associated (i.e., when a MN moves beyond the range of a PoA into a range of another) a PoA change can occur. From the perspective of the network topology, the movement of a MN across the network can be seen as a change in its PoA to the network. This procedure is called handover (also named as handoff) and involves several issues highlighted in [5]:

- How to find the MN's current location?
- How is the new address gained so that data can be forwarded to the MN at the new location?
- How to maintain uninterrupted communications?

Therefore, the main goal of mobility aims to answer all these questions and to enable the MN to be capable of changing its PoA without losing its ability to communicate. This

is specially true in terms of IP address reachability. More specifically, it aims to guarantee continuous and seamless connectivity while the MN changes its position and, consequently, its PoA.

2.2.1 Common Mobility Approaches and Enhancements

In IP networks, an IP address is used as an identifier (ID) and as a locator (LOC). Thus, a PoA is uniquely identified by its IP address and, therefore, a node must be located on the PoA network in order to be able to receive messages destined to it. In situations that involves fixed environments this may make no problem, but the situation is different in mobile environments. Before the definition of mechanisms to support the mobility of a MN, there were two approaches that enabled the MN to be able to communicate after changing its PoA [6]:

- The node must change its IP address whenever it changes its PoA.
- Host-specific routes must be propagated through the Internet.

Each one of these approaches has problems that makes it unacceptable. The first does not allow the MN to maintain its transport and higher-layers connections when it changes its PoA and therefore, does not provide continuous and seamless connectivity. The second has major problems related to scalability.

2.2.1.1 IPv4 Mobility

In order to solve the connection loss and/or scalability problems, several mobility management protocols were defined. MIP [6] is one of the first and still the main mechanism developed by the Internet Engineering Task Force (IETF) to support node mobility. The essence of MIP lies on identifying each node by a home address that is independent of its current location (i.e., on an ID-LOC separation approach). Thus, the MN will have two IP addresses: one in the home network (as an ID), called home address, and the other in the visited network (as a LOC), called Care-of Address (CoA). This enables the MN to change its PoA and still continue to be identified by its home address allowing seamless connection, while the MN changes its PoA. MIP is also independent of the access technology of the current and the new PoA, i.e., it is just as suitable for mobility across homogeneous media as it is for mobility across heterogeneous media.

This mechanism introduces the following new functional entities to the network architecture: the Home Agent (HA) and the Foreign Agent (FA). While the HA is a router on the home network of the MN that maintains the current location of the MN and tunnels the messages to it when it is away from its home network, the FA is a router on the visited network that routes the messages to the MN while registered.

The basic operation of this mechanism (Figure 2.4) is based on the assignment of a CoA to the MN when roaming, and the establishment of tunnels and specific route update mechanisms that forward the messages from the home network to the visited network. When the MN is at the home network, the packets to and from it will be routed by using standard IP routing. If the MN roams to a foreign network, the MN will register its new CoA with its HA to inform about its new location (called the binding procedure). Starting at this time, the messages sent to the MN will be intercepted by the HA and tunnelled to the FA, that finally routes them

to the MN identified by its CoA. For messages sent by the MN there is no need to implement new routing mechanisms since the route of the messages to the destination is made according to conventional IP mechanisms. This indirect routing approach is also known as triangular routing.

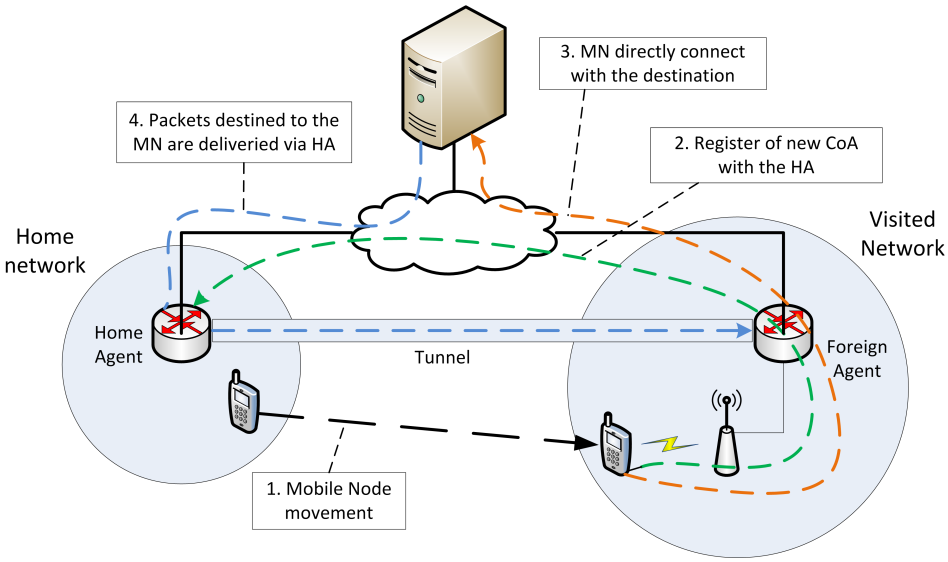

Figure 2.4: MIP routing

However, several problems were detected [9] [10], such as the significant increase of network overhead in terms of delay, packet loss and signalling. Thus, several extensions and enhancements were proposed.

Regional registration [11] defines a new registration mechanism to reduce the number of signalling messages to the home network and, therefore to reduce the signalling delay when a MN moves from one FA to another within the same visited network. In addition to this new extension, IP paging support [12] allows the minimization of the signal overhead and also has the benefit of preserving the power reserves of mobile hosts. This also optimizes mobility management performance on local mobility. Route optimization procedures [13] can improve service quality since it minimizes the effects induced by the triangular routing. This procedure allows the messages destined to the MN to be routed from the origin node to the MN without going to the HA first.

2.2.1.2 IPv6 Mobility

Support for mobility in IPv6 is defined in MIPv6 [2], which takes advantage of the features provided by IPv6 addresses. Although MIPv4 is based on MIPv6, and therefore shares the same bases, there are some major differences:

- No need to deploy the FA as in MIPv4. MIPv6 operates in any location without any added support required from the local router, since it relies on basic IPv6 procedures.
- Support for route optimization (Figure 2.5), unlike MIPv4 which it is optional.
- Provides neighbour discovery mechanisms independently of the link layer technology since it uses IPv6 Neighbour Discovery instead of Address Resolution Protocol (ARP).

It also ensures symmetric reachability between the MN and its default router in the current location.

- The use of IPv6 encapsulation and the routing header removes the need to manage the soft state of the tunnel. It also enables the MN to send a message while away from the home network, by using IPv6 routing instead of IP encapsulation, reducing overhead when compared with MIPv4.

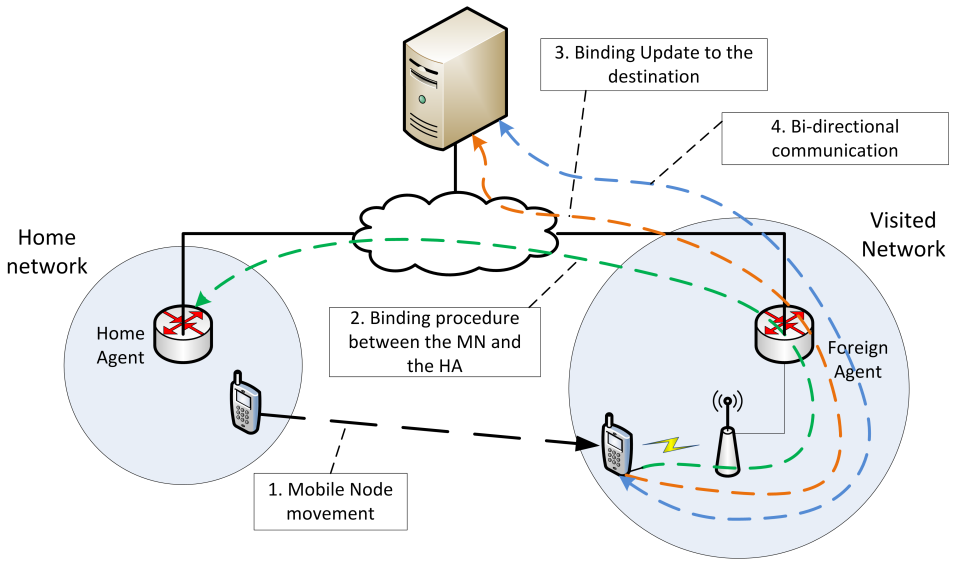

Figure 2.5: MIPv6 routing

2.2.1.3 IP Mobility Extensions

As in MIPv4, MIPv6 procedures (more specifically link-layer procedures, movement detection, IP address configuration, and location update) cause delay in the handover process, preventing the MN to send or receive messages during this period. This handover latency is often sufficient to affect real-time communications. Thus, to reduce handover delay and packet loss the IETF specified extensions to the standard MIPv6: Fast Handover Mobile IPv6 (FMIPv6) [14] and Hierarchical Mobile IPv6 (HMIPv6) [15].

FMIPv6 is an extension that allows an access router (AR) to offer services to the MN in order to anticipate L3 handover. It allows the reduction of the handover latency and packet loss by anticipating the L3 handover via L2 triggered information, i.e., when MN detects a possible L3 handover, it can connect to a new AR while connected to the old AR (if the MN has more than one interface) and it can instruct the old AR to forward the messages to the new AR which is then responsible to forward them to the MN.

HMIPv6 aims to reduce latency and signal overhead of MIPv6 during the binding update phase. In order to achieve its purpose, it introduces a new entity called Mobility Anchor Point (MAP), which is an entity located on the visited network that works like a local HA. When a MN enters a MAP domain, two addresses will be assigned: Regional CoA (RCoA), which is used by the MN to inform the HA and destination nodes about its current location, and a Local CoA (LCoA), which identifies the MN location inside the MAP domain. Thus, when

the MN is moving inside the MAP domain, the RCoA does not change and therefore it does not need to inform its HA or destination nodes, minimizing the latency and signal overhead.

The integration of several mechanisms described previously as well as extensions to them have been widely studied ([16], [17], [18], to name a few). They aim to improve the performance of the handover and to decrease the footprint left by it.

2.2.1.4 Network Based IP Mobility

The approaches mentioned previously are all host based, i.e., the MN must signal themselves to the network when their location changes and must update routing states in the HA. However, the increase in types of access technologies and the diversity and size of offered services by the access network, as well as the increasing number of multi-access capable terminals equipped with IP technologies, create complex environments. Furthermore, the information required for the selection of the target network can depend on policies and commercial roaming arrangements on the access network, access provider and service provider levels. Some of this information is only available to network elements and therefore, the MN might not have enough information to make an intelligent handover decision [19]. Thus, network based approaches have been studied and developed, particularly by the IETF. So, the IETF developed a network based mobility management protocol named PMIPv6 [3]. This protocol aims not to involve the MN in any IP layer mobility related signalling. Instead, it aims to provide mobility to the MN without its direct involvement in related signalling. The mobility entities in the network are responsible for tracking the movement of the MN and will initiate the mobility signalling and set up the required routing states. Thus, PMIPv6 introduces two new network functional entities, which are responsible for managing IP addresses involved in the MN mobility:

- **Mobile Access Gateway (MAG):** performs the mobility management on behalf of the MN. It resides on the access link where the MN is anchored.
- **Localized Mobility Anchor (LMA):** manages and maintains the reachability towards the MN through the creation of tunnels directed to one of several MAGs, constituting a PMIPv6 domain.

The approach taken by PMIPv6 is similar to HMIPv6, where the mobility management is made in an hierarchical way, reducing the amount of signalling outside the local domain. The MAG and the LMA entities are compared with the AR and MAP entities of the HMIPv6, respectively.

When the MN first enters a new Local Mobility Domain (LMD) and attaches to an access network (Figure 2.6), the MAG on that link detects the attachment of the MN. Then the MAG will register the MN with the LMA through a Proxy Binding Update (PBU) message in order to associate its own address with the identity of the MN. Upon the reception of this request, the LMA allocates a home network prefix that will be assigned to the MN, and informs the MAG by sending a Proxy Binding Acknowledgement (PBA) message. It also establishes a bidirectional tunnel to the MAG. Finally, the MAG advertises the assigned home network prefix to the MN in an unicast Router Advertisement. From then on, once the MN moves within the same LMD and changes its PoA from one MAG to another, the new MAG updates the location of the MN in the LMA and advertises the same prefix to the MN (Figure 2.7). For communications outside its domain, the LMA works as a HA.

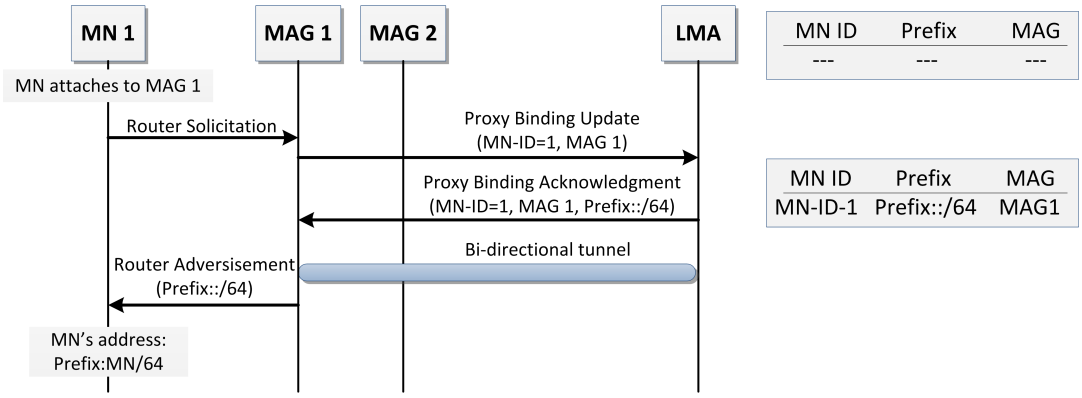

Figure 2.6: PMIPv6 signalling when the MN connects to a PMIPv6 domain

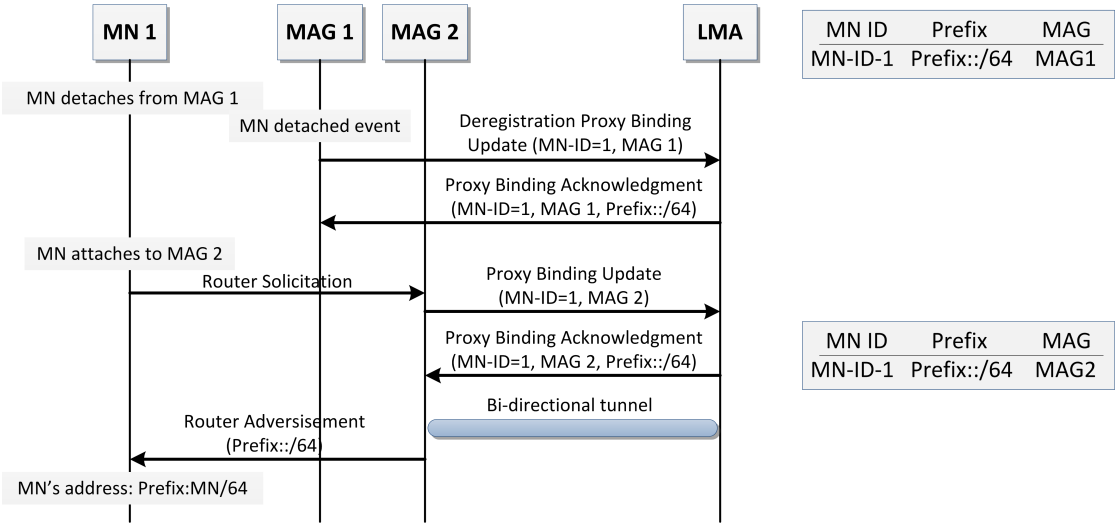

Figure 2.7: PMIPv6 signalling when the MN change its PoA

However, several link-layer mechanisms which are important to PMIPv6 operations, such as handover candidate detection and selection, network resources querying and committing and network attachment detection, are out-of-scope of this standard.

2.2.2 Discovery mechanisms

Among the mobility mechanisms depicted previously, several discovery procedures have to be defined to support mobility scenarios. These procedures have been the subject of extensive research work and aim to provide flexibility when the MN does not have enough information about the network, as well as to optimize the mobility mechanisms by allowing automatic entities and/or services discovery. It is also of great importance to enable mobility related protocols to speed up handover.

For example, before the MN can exchange MIPv6 signalling with a HA, it requires the foreknowledge of its connectivity parameters (i.e., IP address, port and protocol). Then, if the MN reboots on a foreign network, it must somehow discover information about its HA, otherwise it won't be able to communicate with it. The MN can have the IP address of

the HA pre-configured or it can discover the HA dynamically. Thus, MIPv6 [2] defines a mechanism to dynamically discover the HA address named Dynamic Home Agent Address Discovery (DHAAD), which is depicted in Figure 2.8. When the MN needs to know the HA address it sends an ICMP Home Agent Address Discovery Request message to the MIPv6 HAs anycast address for its home network IP subnet prefix. Upon the reception of this message, a HA replies with a ICMP Home Agent Address Discovery Reply message to the MN with the source address of the reply packet set to one of the global unicast addresses of the HA. The MN, by receiving this message, can discover its HA and register with it.

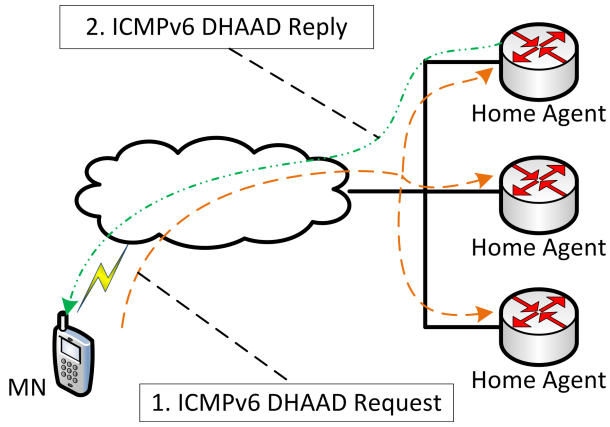

Figure 2.8: Dynamic Home Agent Address Discovery procedure

However, this procedure involves the foreknowledge of the home network prefix. To overcome this, two solutions were proposed involving Dynamic Host Configure Protocol (DHCP) [20] or the AR [21] in the discovery process. The first specifies new DHCP options that carry the HA information, such as home network prefix, home agent IP address and Fully Qualified Domain Name (FQDN) information. The MN can request this information by including the DHCP options defined in an Information-request message according to the stateless DHCPv6 procedures [22] [23]. Then, the MN can use that information to discover the list of HAs by using the DHAAD mechanism. The second solution is based on the enhancement of Router Solicitation and Router Advertisement messages with additional options. After the establishment of a L2 connection between the MN and the AR, the MN sends a Router Solicitation message to the AR that includes a Network Access Identifier (NAI) identifier. The AR replies by sending a Router Advertisement that includes the Home Subnet Prefix or a list of HA addresses.

Other mechanisms, such as the MAP discovery defined in HMIPv6 [15], also make use of ARs to discover mobility entities. MAP discovery defines how the MN discovers the MAP address and subnet prefix. The MN must be aware of the Router Advertisement messages and, upon their reception, it searches for the MAP options that are encapsulated within these messages. Based on the received information the MN can choose to which MAP it will register with. Enhancements to provide AR transparency were also subject of study [24].

To make the handover seamless the MN needs to have prior knowledge about the ARs to which it can connect. Thus, to enable the MN to discover candidate access routers (CARs) a discovery protocol named CAR Discovery (CARD) [25] was developed. This protocol enables the discovery and the acquisition of information about the ARs that are candidates for the MN to connect. The MN can discover new ARs by listening to the L2 identifier of one or

more APs. Subsequently, through this identifier, the MN can discover the IP address of the associated CAR that connects to the AP. Additionally, the CARD protocol enables the MN to discover the capabilities associated with the CARs, that might affect the handover decision.

In cellular networks, the 3GPP (3rd Generation Partnership Project) standadization body introduces the Access Network Service Discovery Function (ANSDF) [26]. This logical entity was added to the system architecture to facilitate discovery procedures, enabling the discovery of either 3GPP or non-3GPP target access networks that best fit the requirements of the MN. It also aims to minimize the impact of the use of radio signals by providing information about neighbour cells (such as QoS capabilities which cannot be distributed by radio signals due to high data demand) [27]. In this way, the MN does not need to have multiple interfaces active for discovering new access networks, reducing power consumption, as well as, becoming able to discover information about neighbour cells even if they are not broadcasting.

The ANDSF can provide the following types of information [28]:

- **Inter system mobility policy:** contains a set of operator-defined rules and preferences that have an impact on the handover decision.
- **Access network discovery information:** provides a list of available access network, including the access type technology, a radio access network identifier and other technology specific information.
- **Inter system routing policy:** provides inter system routing policies to MNs that are capable of routing IP traffic simultaneously over multiple radio access interfaces.

[29] proposes an enhancement to the information retrieved by the ANDSF server, focusing on energy efficiency. It also proposes a way to provide the information in a push mode. Figure 2.9 presents the two operating modes that allow the MN acquire information from the ANDSF server.

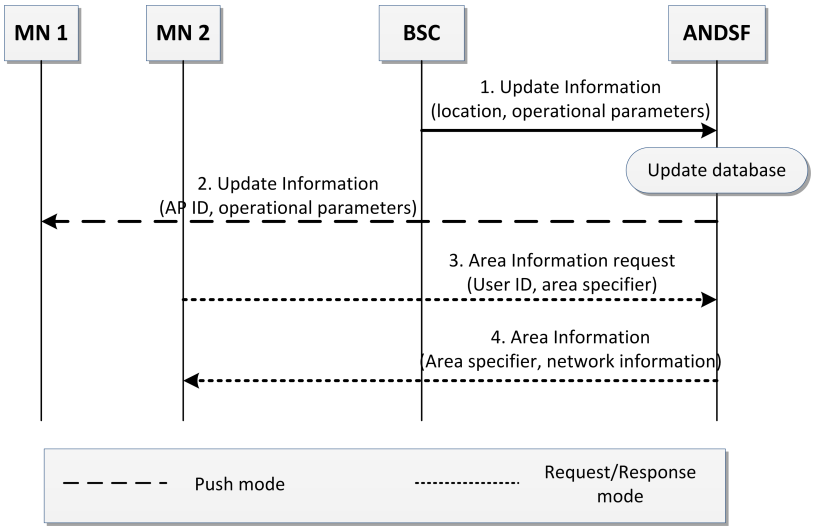

Figure 2.9: Signalling for retrieve information to the MN

In push mode, the MN subscribes to the reception of certain information with the ANDSF server. Then, when the subscribed information changes or new contents (step 1) become available, the ANDSF server is responsible for pushing that information out to the MN that

subscribes it, without any participation by the MN (step 2). The MN can also directly query the ANDSF server for network information on a given area by sending a request message specifying the area that it wants to receive network information (step 3). The ANDSF server is then responsible to retrieve the network information from that area to the MN (step 4).

The communication between the MN and the ANDSF server is IP based, and therefore the MN must know about its existence by discovering the IP address or the FQDN of the ANDSF server. Thus, [26] defines new attributes to facilitate ANDSF bootstrapping via a Remote Authentication Dial In User Service (RADIUS) infrastructure. When the MN attaches to an access network, the AR will require authentication and authorization, which, in some cases, takes place via RADIUS infrastructure. Thus, during the Authentication, Authorization and Accounting (AAA) phase, the MN authenticates and gets authorization to access the services provided by the network. The AAA server provides the AR a list of IP address or FQDN that identifies the ANSDF servers that the MN can access and the list of services they are able to provide. Finally, the AR is responsible to provide the MN such information.

2.3 Summary

This chapter provided an overview of the two different types of wireless technologies that can be used in heterogeneous handovers: IEEE 802.11 and 3G. Here, the main protocols that have been developed to overcome the IP mobility issues were also described and how they evolved in terms of speed, packet loss and handover control. The mechanisms used by mobility related protocols to discover entities and mobility services were also presented in this chapter.

In the next chapter, a study of the services and mechanisms of the IEEE 802.21 standard [7] is presented, which was developed to support media independent handovers.

Chapter 3

The IEEE 802.21 Media Independent Handover Standard

This chapter provides a brief overview of the IEEE 802.21 standard [7], highlighting the aspects that are more closely related to this work.

3.1 Definition

802.21 is an IEEE standard, whose main purpose is to define extensible media access independent mechanisms that may facilitate handovers between heterogeneous technologies, including IEEE 802 (both wired and wireless) and cellular technologies. In addition, it also encompasses aspects related to handover optimization by providing abstract means for mobility management entities to adapt the different link technologies to provide the necessary data rate or to trigger a handover if this rate is not available on the current link. More exactly, this standard aims to enhance handover procedures with abstract link-layer information from both the terminal and network.

This standard copes with mechanisms to support MN-initiated, network-initiated, MN-controlled and network-controlled handovers. Due to the movement of the MNs, changes in link conditions can occur, such as loss of wireless coverage, that require a handover. It can also occur cases where the surrounding environment changes, requiring a handover to a network more suitable than the current one. Thus, the 802.21 standard provides support for handovers for mobile or static entities.

The main design elements of IEEE 802.21 can be classified into four categories:

- A framework that facilitates seamless handover between heterogeneous link-layer technologies. This framework can properly identify the mobility-management protocol stack residing in the network elements that support the handover;

- A set of handover-enabling functions within the protocol stacks of the network elements and the introduction of a new logical entity called MIH Function (MIHF);

- A media independent handover Service Access Point (SAP), called the *MIH_SAP*, and the associated primitives, enabling the MIHF to provide services to upper layers;

- The definition of media specific SAPs and associated primitives for each link layer

technology in order to provide the MIHF access to services from lower layers. These can be grouped in a *MIH_LINK_SAP* representation.

In order to operate and support the 802.21 mechanisms, each involved host and network entity will include a MIHF as a cross-layer function within their protocol stack. The MIHF supplies services to upper layers through the *MIH_SAP* and obtains services from lower layers through media specific SAPs. The MIH entities that belong to the upper layers are also called MIH-Users.

The services provided by this standard are:

- **Media Independent Event Service (MIES):** provides event classification, filtering and reporting, corresponding to dynamic changes in link characteristics, status and quality.
- **Media Independent Command Service (MICS):** provides methods to configure and control link behaviour related to mobility and handovers.
- **Media Independent Information Service (MIIS):** provides mechanisms to acquire, store and retrieve information about networks in the coverage area.

Communications between MIHFs from different entities, as well as provision of the MIH services, facilitates addressing some factors that affect handovers such as: service continuity, application class, QoS, network discovery and selection, power management and handover policy.

3.2 General Architecture

This section gives a brief overview of the IEEE 802.21 architecture. It describes the communication model, its main entities, the services provided and the protocol used for remote communications.

3.2.1 Communication Model

MIHFs from different entities can communicate with each other for various purposes, depending on its function in the network. Thus, the MIHF can be present on the following network entities:

- The MN, which is the communicating node that can change its PoA from one link to another.
- The Point of Service (PoS), that is a network side entity that provides MIH services to the MN.
- The PoA, that is the network side endpoint of a L2 link to which the MN connects to. The PoA can be collocated with PoS.

The MIHF communication model (Figure 3.1) shows the different roles of each entity and the communication relationship between them. It is possible to identify five types of relationships as described in Table 3.1.

Further detais regarding the mapping between the different MIH messages to each one of the reference points can be obtained in the IEEE 802.21 standard [7].

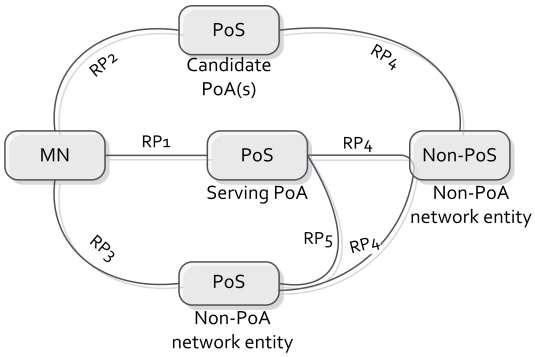

Figure 3.1: MIH communication model

Table 3.1: Reference Points

Reference Point	Transport	Description
RP1	L2, L3 and above	Communications between a MN and a PoS in the serving PoA network entity.
RP2	L2, L3 and above	Communications between a MN and a PoS in the candidate PoA network entity.
RP3	L2, L3 and above	Communications between a MN and a PoS in the non-PoA network entity.
RP4	L3 and above	Communications between a PoS and a non-PoS entity in different network entities.
RP5	L3 and above	Communications between two PoS entities in different network entities.

3.2.2 The Media Independent Handover Function

The MIHF is the central entity of the IEEE 802.21 standard, defined as a logical entity whose purpose is to facilitate handover decisions to the network selector entity. Its purpose is achieved by providing the inputs to higher layers through abstract services and communicating with the lower layers of the mobility-management protocol stack through technology-specific interfaces. By other words, the MIHF provides mechanisms that enable the upper layers to manage and control the handover procedure in a media independent way. Figure 3.2 illustrates the MIHF regarding its placement between the upper and lower layers.

The MIHF can be seen as a cross layer interfacing all layers between the Data Link Layer and the Application Layer from the mobility-management protocol stack, since it enables directly communications from the Data Link Layer to any of the upper layers, for handover control and optimization procedures.

3.2.3 Service Access Points

The SAPs are a set of primitives that enable the MIHF to be access by the MIH-Users to communicate with other functional planes. These SAPs can be categorized in two groups: the media independent and the media dependent ones. The first allow the MIHF to provide services to the upper layers, while the second allow the MIHF to use services from lower layers.

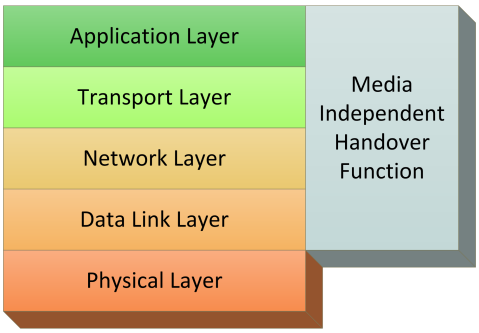

Figure 3.2: MIHF position in the protocol stack

The primitives can be divided in four groups: request, confirm, indication and response. The request primitive is used by the layer that wants to get services from other layers and the confirm message is the correspondent response to that request. The indication message is the notification that another layer requests a service and the response message is the acknowledgement of that request.

Figure 3.3 illustrates the position of the MIHF and its interactions with other entities.

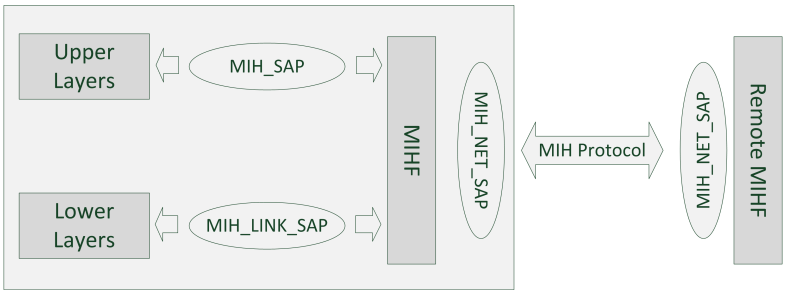

Figure 3.3: SAPs relationships

The media independent SAPs provide services to upper layers through a single interface, named *MIH_SAP*. This interface provides functions to the upper layers that need to subscribe with the MIHF in order to receive events generated by the MIHF or by the layers below the MIHF. It also provides functions to upper layers enabling them to send commands to the MIHF, to control the link layers in a abstract way.

The IEEE 802.21 standard defines an abstract interface for communications between the MIHF and lower layers, called *MIH_LINK_SAP*. Since the media dependent SAPs depend on the link layer technology, each link layer technology specifies its own media dependent specific SAP.

In order to provide transport services over the data plane on the local node and communications between remote MIHF entities, the IEEE 802.21 standard defines a set of primitives grouped in *MIH_NET_SAP*. It makes use of the MIH protocol, which was defined for transporting messages between a pair of MIH entities.

3.2.4 MIH Services

The MIH defines a set of services that aim to help MIH-Users to manage, determine and control the state of underlying interfaces. These services can be delivered asynchronously or

synchronously through well defined SAPs for upper and lower layers.

3.2.4.1 Media Independent Event Service

The MIES is responsible for notifying the upper layers about events from the lower layers, which can be originated from local or remote interfaces. These events are sent asynchronously to the MIHF, indicating changes in the state and the transmission behaviour of the physical, data link and logical link layers, predictions on state changes of these layers and changes in dynamic link characteristics such as link status and link quality. More specifically, the MIES aims to be used to detect the need for handover.

Events can have origin in the MIHF, called MIH Events, or in lower layer, called Link Events. The typical flow of events has its origin at the lower layers and it is sent to the MIHF and, if a subscription for these events is made, this flow continues to higher layers entities.

Figure 3.4 shows the two possible event service flows, i.e., local and remote event flows. Local events are often propagated from the lower layers to the MIHF and from the MIHF to any upper layer, whereas remote events are propagated from the MIHF in one protocol stack to the MIHF in the peer protocol stack.

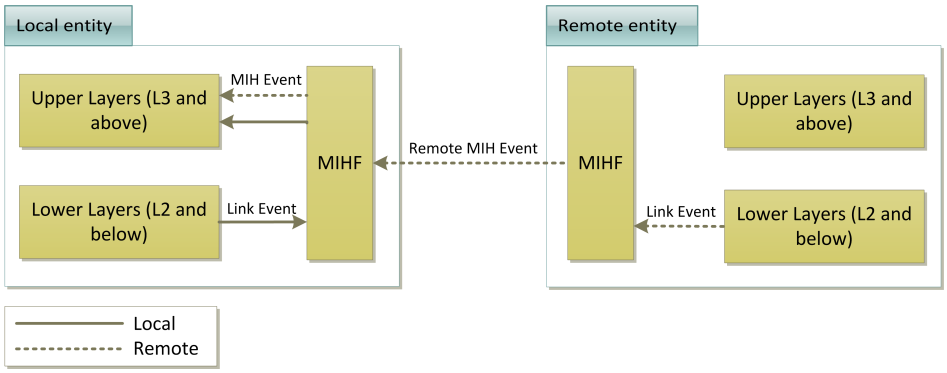

Figure 3.4: Local and Remote Events

In order to receive event notifications, both from local lower layers or from remote entities, MIH entities must subscribe to receive such notifications. The subscription can be issued by the MIHF with the lower layers, called link event subscription, or can be issued by upper layers with the MIHF, called MIH event subscription. It is also possible for multiple entities to subscribe to the same event, in which case all subscribers receive the event notification.

The MIES supports several types of events:

- **MAC and PHY State Changes events:** concern about definite changes in MAC and PHY state, such as attachment and detachment events.

- **Link Parameter events:** triggered by changes in the link layer parameters, such as signal strength, throughput or data rate. They can be generated in a synchronous way (in response to a request from upper layers) or asynchronously (by reporting when a specific parameter reaches a threshold).

- **Predictive events:** which correspond to the probability of future changes in the link conditions based on past and present conditions.

- **Link Handover events:** report to the higher layers about the occurrence of L2 handovers.
- **Link Transmission events:** give indications about the link layer transmissions status. This information can be used by the upper layers to improve buffer management in order to achieve minimizations of data loss.

Table 3.2 presents the current link events defined in the IEEE 802.21 standard, both for MIH events sent from the MIHF to the MIH-Users, and MIH events sent from the link layers to the MIHF.

Table 3.2: 802.21 MIES primitives

Primitive Name	**Event Type**	**(L)ocal (R)emote**	**Description**
Link_Detected MIH_Link_Detected	State Change	L, R	L2 detection of a new link.
Link_Up MIH_Link_Up	State Change	L, R	L2 connection is successfully established and the link is available for use.
Link_Down MIH_Link_Down	State Change	L, R	L2 connection is lost and the link is not available for use.
Link_Parameters_Report MIH_Link_Parameters_Report	Link Parameters	L, R	Indicate changes in link parameters that have crossed a prespecified threshold.
Link_Going_Down MIH_Link_Going_Down	Predictive	L, R	L2 link is losing connectivity.
Link_Handover_Imminent MIH_Link_Handover_Imminent	Link Handover	L, R	L2 handover decision is complete and its execution is imminent.
Link_Handover_Complete MIH_Link_Handover_Complete	Link Handover	L, R	L2 link handover has been completed.
Link_PDU_Transmit_Status MIH_Link_PDU_Transmit_Status	Link Transmission	L, R	Indicate the transmission status of a PDU.

As an example of use, event services are helpful to detect when a handover is needed. Lower layers can report that the current link L2 connection loss is imminent through analysis of the signal level. The upper layers, by receiving this event, can prepare and initiate the handover to a new PoA.

3.2.4.2 Media Independent Command Service

The MICS allows higher layers to configure, manage, control and get information from lower layers. Through the command service it is possible to determine the status of links and to control the device, aiming to manage and control link behaviour relevant to handovers and mobility. The information retrieved by the command service is characterized by dynamic information composed of link parameters such as signal strength and link speed.

These commands can be requested by the upper layers, called MIH Commands, or can be requested by the MIHF, called Link Commands. The typical flow of commands has its origin at the upper layers and it is sent to the MIHF, which is then sent to the lower layers.

However, commands can also be originated by the MIHF with no interference of the upper layers. Figure 3.5 represents the possible flows of command messages.

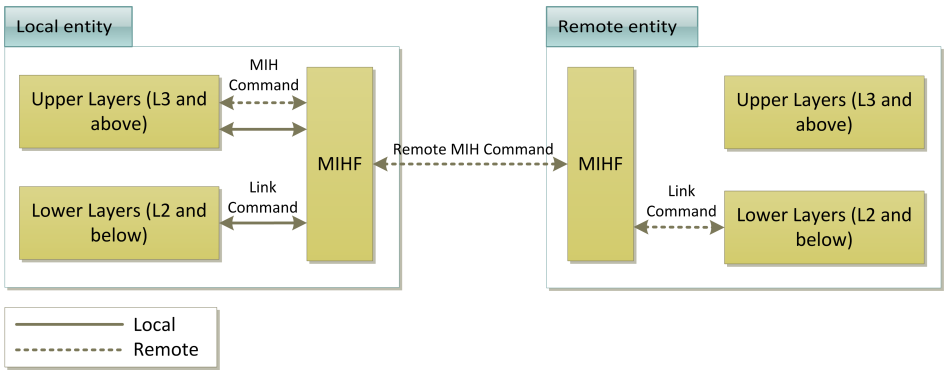

Figure 3.5: Local and Remote Commands

MIH and link commands may be local or remote. Local commands are sent by upper layers to the MIHF in the local stack, whereas remote commands are sent by upper layers to the MIHF in the peer stack.

Table 3.3 presents the current link events available in the 802.21 standard.

As an example of use, command services can be used to request a MN to initiate the handover procedure. The MN uses the set of *MIH_MN_HO_**** commands to query the list of available candidate networks, reserve the necessary resources at the selected target network and indicate the status of handover operation to the network.

3.2.4.3 Media Independent Information Service

The MIIS provides a framework by which a MIHF entity can discover and acquire network information within a geographical area in order to facilitate handovers, similar to the ANDSF of the 3GPP. Contrary to dynamic information obtained through MICS, information obtained via MIIS is mostly static. The network information can either be related to the same access technology that the entity is using to request the information and/or be related to other type of access technologies. For example, by using an IEEE 802.11 access network a MN gets information not only about all other IEEE 802 based networks in a geographical area but also about 3GPP and 3GPP2 networks. Thus, the main purpose of the MIIS is to allow the MN and network entities to discover information, such as knowledge of security information, supported channels, cost per use, networks categories and QoS supported, that helps in the selection of appropriate networks during handovers.

The MIIS provides one single primitive for local and remote communications (Figure 3.6), named *MIH_Get_Information*.

The information may be stored in a MIIS server where the requestor may access it. In order to exchange this information between the MIIS server and the requestor, the MIIS provides a set of Information Elements (IEs), the information structure and its representation, as well as a query/response mechanism.

The information carried in MIIS messages can be classified in three groups:

- **General and Access Network Specific Information:** information about the general overview of the different networks within a geographical area. A list of available

Table 3.3: 802.21 MICS primitives

Primitive Name	**(L)ocal (R)emote**	**Description**
Link_Capability_Discover	L	Query and discover the list of supported link layer events and commands.
Link_Event_Subscribe	L	Subscribe to a set of events from a link.
Link_Event_Unsubscribe	L	Unsubscribe a set of events from a link.
Link_Get_Parameters MIH_Link_Get_Parameters	L, R	Get the status of a link.
Link_Configure_Thresholds MIH_Link_Configure_Thresholds	L, R	Configure thresholds of a link parameter.
Link_Action MIH_Link_Action	L, R	Invoke actions to control the behaviour of link layers.
MIH_Net_HO_Candidate_Query	R	Initiate handover by the network and send a list of suggested networks and associated PoAs.
MIH_MN_HO_Candidate_Query	R	Query executed by the MN to get handover related information about possible candidate networks.
MIH_N2N_HO_Query_Resources	R	Sent by the serving MIHF entity to the target MIHF entity to allow for resource query.
MIH_Net_HO_Commit	R	Used by the network to inform the MN about the selected target network.
MIH_MN_HO_Commit	R	Used by the MN to notify the serving network with information about the selected target network.
MIH_N2N_HO_Commit	R	Used by the serving network to inform the target network that a MN is about to handover to it.
MIH_MN_HO_Complete	R	Used by the MN to notify the target or source network about the status of the handover completion.
MIH_N2N_HO_Complete	R	Used by the target or source network to notify a peer entity about the status of the handover completion.

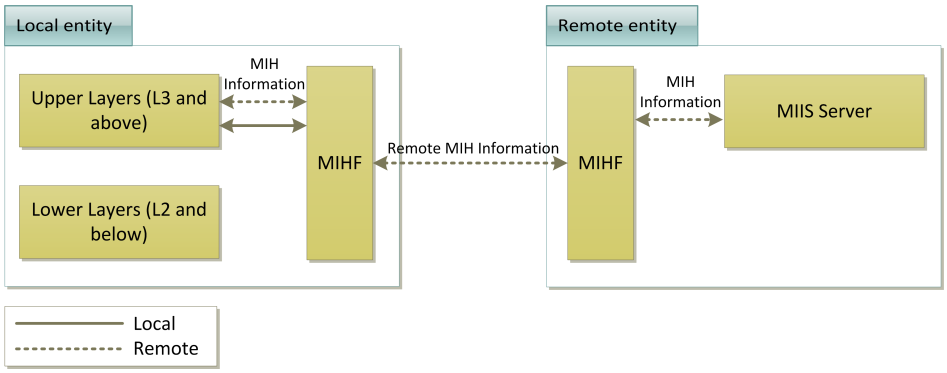

Figure 3.6: Local and Remote Information Exchange

networks, and their associated operator, are an example of this type of information.

- **PoA Specific Information:** information about PoAs for each of the access networks available. This information can include lower layers information, such as address, location, data rate and so on, or can include higher layer services and individual capabilities of each PoA.
- **Other information:** access network, service or vendor/network specific information.

The IEs can be represented either by a binary representation or by the Resource Description Framework (RDF) representation. If binary representation is used, a Type-Length-Value (TLV) query method must be used. If RDF representation is used, the SPARQL Protocol and RDF Query Language (SPARQL) query method must be used.

3.2.4.4 Service Management

In order to use MIH services from one MIHF, MIH entities need to be properly managed and configured by using a fourth intrinsic service called service management. This service provides mechanisms for discovering capabilities, registering and subscribing events of a MIHF through a set of primitives defined in Table 3.4.

Table 3.4: 802.21 Service Management primitives

Primitive Name	**(L)ocal (R)emote**	**Description**
MIH_Capability_Discover	L, R	Discover MIHF's capabilities.
MIH_Register	R	Register with a remote MIHF.
MIH_DeRegister	R	Deregister from a remote MIHF.
MIH_Event_Subscribe	L, R	Subscribe to a set of events from a MIHF.
MIH_Event_Unsubscribe	L, R	Unsubscribe from a set of events from a MIHF.

The upper layers can discover local or remote MIHF capabilities, more specifically the MIH services supported, by using the MIH Capability Discover procedure. This procedure can be transmitted either through the MIH protocol or through media specific mechanisms (for WLAN the IEEE 802.11 Beacon frames or the IEEE 802.11 Management frames can be used). The MIH Capability Discover main purpose is to gather information from peer MIHFs in order to decide and select the MIHF to register with.

After being chosen the MIHF that will provide the MIH service, it is necessary to request the access to these services by using the MIH Register mechanism. This mechanism is particularly mandatory to use the MIH Command Service and the push mode of the MIH Information Service.

Like the event subscribe procedure of the MICS, the MIH Event Subscribe procedure allows the upper layers to subscribe for a particular set of events that originates from a local or remote MIHF.

3.2.4.5 Deployment examples for the MIH services

This section will describe different possible deployment scenarios taking in consideration the position of the MN and the PoS [30]. Figure 3.7 illustrates some deployment scenarios for the MIH services and how the MN can access them.

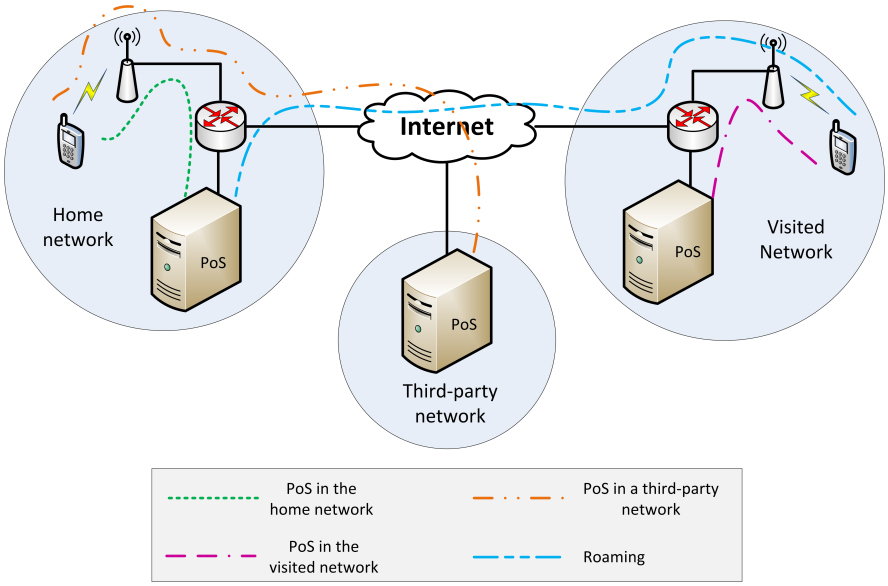

Figure 3.7: Deployment of the MIH services

Based on the position of the MN and the PoS there are the following possibilities:

- **PoS in the home network:** In this scenario, the MN and the PoS that provides the MIH services to the MN are located in the home network. The PoS can be located in the same access network of the MN or somewhere else.
- **PoS in the visited network:** In this scenario, the MN is in the visited network and it accesses the MIH services provided by the visited network.
- **PoS in a third-party network:** In this scenario, the MN is in the home or visited network and it accesses the MIH services located on a third-party network.
- **Roaming:** The last scenario refers to situations where the MN is in the visited network and pretends to access to the MIH services provided by its home network.

The position of the MN affects how the discovery procedure should be done. This is discussed in a later chapter of this work.

3.3 The Media Independent Handover Protocol

The communication between peer MIHFs is essential to enable the optimization of handovers between heterogeneous networks. So, in order to enable the MIHFs to remotely exchange messages, the MIH protocol was defined. This protocol defines the format of the messages exchanged between peer MIHF, as well as the mechanisms that support their transport and reliable delivery of the messages to the destination [31].

3.3.1 Protocol Identifiers

The communication between peer MIHF entities requires the proper identification of the transaction. Thus, the MIH protocol provides two identifiers that uniquely identify the

transaction: the MIHF identifier (MIHF ID) and the transaction identifier (Transaction ID).

The MIHF ID is an identifier that uniquely identifies a MIHF entity and that is required for the correct deliver of the messages. This identifier may be assigned during the MIHF configuration process and must be represented in the form of a Network Access Identifier (NAI) [32]. In addition, [30] proposes that this identifier can also be represented in the Fully Qualified Domain Name (FQDN) [33] form. The MIH protocol also defines a multicast MIHF ID which corresponds to an MIHF ID of zero length. This is usually used when the MIHF ID of the destination is unknown.

The Transaction ID is an identifier that is used to match a request message with its corresponding response message, as well as for matching request, response or indication message with the respective acknowledge message. A Transaction ID must be unique among all the pending transactions between two peers entities and it is created at the MIHF that starts the transaction.

3.3.2 Frame Format

The MIH protocol message is composed by MIH protocol header (Table 3.5) and the MIH protocol payload. The payload carries two mandatory TLVs that identify the source and the destination MIHF entities, followed by MIH service specific TLVs. Figure 3.8 presents the organization of the various components in the MIH message.

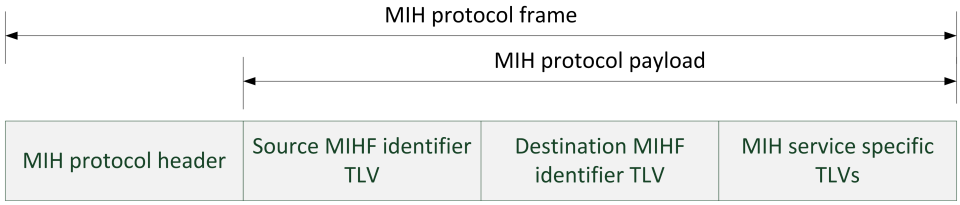

Figure 3.8: MIH protocol frame generic format

The MIH protocol header carries the essential information that is presented in every frame and which is used for parsing and analyzing the MIH protocol frame. The payload of the MIH message consists of a set of TLVs which carry the message parameters. Figure 3.9 represents the TLV format.

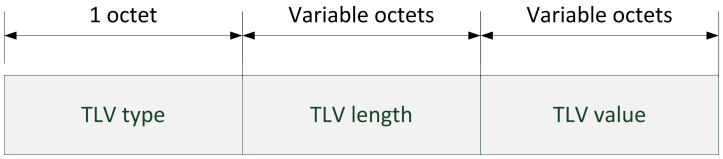

Figure 3.9: TLV format

The messages of the MIH protocol are divided in four categories: Service Management, Event Service, Command Service and Information Service (which have been defined in the previous section), as described in Table 3.6.

Table 3.5: Description of MIH protocol header fields

Field Name	**Size (bits)**	**Description**
Version	4	Used to specify the version of MIH protocol used.
ACK-Req	1	Used for requesting an acknowledgement of the message.
ACK-Rsp	1	Used for responding to the request for an acknowledgement of the message.
Unauthenticated information request (UIR)	1	Used by the MIIS to indicate if the protocol message is sent in pre-authentication/pre-association state.
More fragment	1	Used for indicating that the message is a fragment to be followed by another fragment.
Fragment number	7	Used for representing the sequence number of a fragment.
Reserved1	1	Intentionally kept reserved.
MIH message ID (MID) - Service identifier (SID) - Operation code (Opcode) - Action identifier (AID)	16 4 2 10	Used for representing the message identifier and it is a combination of the following 3 fields: - This field identifies the different MIH services. - This field identifies the type of operation to be performed. - This field indicates the action to be taken.
Reserved2	4	Intentionally kept reserved.
Transaction ID	12	Used for matching request and response, as well as matching request, response and indication to an ACK.
Payload length	16	Indicates the total length of the variable payload embedded in the MIH protocol frame.

3.3.3 Transport Considerations

In order to provide flexibility, the IEEE 802.21 standard defines transport mechanisms for lower and upper layers (as shown in Figure 3.10). Thus, the MIH protocol provides the capability for transferring MIH messages between peer MIHFs over the data plane by using suitable transport mechanisms at L2 or L3.

L3 transport is based on IP, using one of the following transport protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP) or Stream Control Transmission Protocol (SCTP).

At L2 level, the MIH messages can be sent over the data plane after the MN authenticates with the access network. However, for IEEE 802.11 and IEEE 802.16 networks, it is possible to send MIH messages (limited to the MIIS Query request/response, MIES and MIH Capability Discover from MICS) before any authentication with the network. This can be achieved by sending MIH messages over the management plane, using media specific MAC management frames.

Since the message delivery between two peer MIHF can be made through an unreliable transport mechanism, the MIH protocol defines an Acknowledgement Service, which provides reliable services. However, this acknowledgement service is optional and it is not needed when

Table 3.6: MIH messages

MIH message	Category
MIH_Capability_Discover	Service Management
MIH_Register	Service Management
MIH_DeRegister	Service Management
MIH_Event_Subscribe	Service Management
MIH_Event_Unsubscribe	Service Management
MIH_Link_Detected	Event Service
MIH_Link_Up	Event Service
MIH_Link_Down	Event Service
MIH_Link_Parameters_Report	Event Service
MIH_Link_Going_Down	Event Service
MIH_Link_Handover_Imminent	Event Service
MIH_Link_Handover_Complete	Event Service
MIH_Link_Get_Parameters	Command Service
MIH_Link_Configure_Thresholds	Command Service
MIH_Link_Actions	Command Service
MIH_Net_HO_Candidate_Query	Command Service
MIH_MN_HO_Candidate_Query	Command Service
MIH_N2N_HO_Query_Resources	Command Service
MIH_MN_HO_Commit	Command Service
MIH_Net_HO_Commit	Command Service
MIH_MN_HO_Complete	Command Service
MIH_N2N_HO_Complete	Command Service
MIH_Get_Information	Information Service
MIH_Push_Information	Information Service

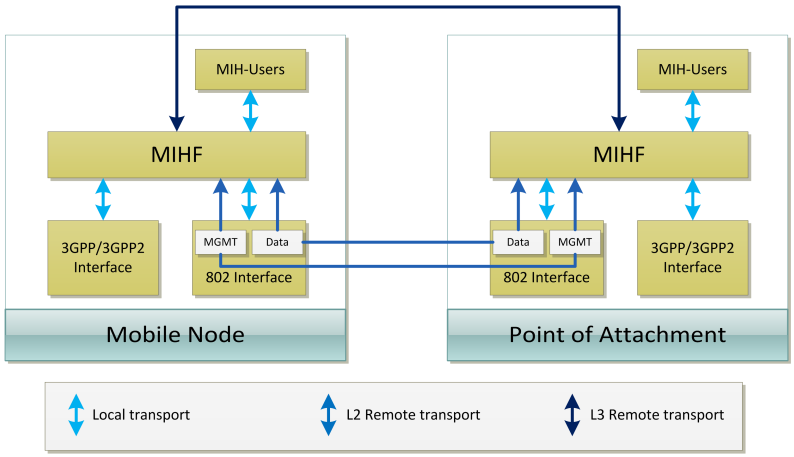

Figure 3.10: MIH protocol message transport

the transport protocol is reliable. The MIH acknowledgement service is supported by the Ack fields (ACK-Req and ACK-Rsp) present in the MIH message header. The ACK-Req bit is set by the source of the MIH message if the requestor expects the acknowledgement from the

message destination. In this way, if the acknowledge message is not received, the source will retransmit the message. The ACK-Rsp is set by the destination and pretends to acknowledge the receipt of the message.

Figure 3.11 depicts the different interactions that can occur with the acknowledge service enabled. When the destination MIH entity receives an MIH messages with the ACK-Req bit set, it must acknowledge its reception by sending a MIH message (with the same TID) with the ACK-Rsp bit set. Usually, the acknowledge message has only the MIH header and no other payload. For indications messages, the behaviour described is also applicable. However, when the destination MIH entity immediately replies to the received message, the ACK-Rsp bit is set in the corresponding response message.

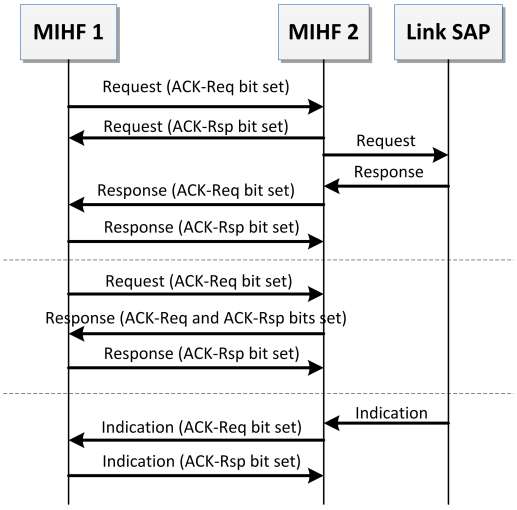

Figure 3.11: MIH protocol acknowledge service

Since the acknowledge service from the MIH protocol retransmits a message that has not been acknowledged (if the ACK-Req bit is set), it can happen that the destination MIH entity will receive duplicated messages. In those cases, the destination MIH entity responds with an acknowledgement message for the duplicates MIH messages that have the ACK-Req bit set. Nevertheless, if the acknowledge bit is not set, the messages are discarded and no actions are taken.

3.3.4 MIHF Discovery

MIHF discovery allows one MIHF to discover peer MIHF entities. One simple example is the ability of a MN to discover available MIHFs in an access network. The IEEE 802.21 standard defines simple directives to enable MIHF discovery either at L2 or L3.

At L2, the MIHF discovery is done either in a media specific manner or by using multicast data frames. The first is achieved by listening to media specific broadcast control messages (e.g., IEEE 802.11 Beacon Frame or IEEE 802.16 Downlink Channel Descriptor) that carry information about the MIHF and its capabilities. This discovery procedure is also known as Unsolicited MIH Capability Discover since there is no implicit request. The second is achieved by combining the MIHF discovery procedure with the MIH Capability Discover. The MIHF that wants to discover new MIH entities can multicast a MIH Capability Discover with MIHF

ID as zero length and, that way it can discover the MIHFs that reside in the same multicast domain. The requestor is also able to discover the capabilities of the discovered MIHF.

At L3, discovery mechanisms are defined at [30] and involves the use of Domain Name System (DNS) [34] and DHCP [35] services.

3.3.4.1 Use of MIH Capability Discover in MIHF Discovery

Figure 3.12 describes an example of the usage of MIH Capability Discover procedure in the process of MIHF discovery. All PoAs and PoS are unknown to the MN.

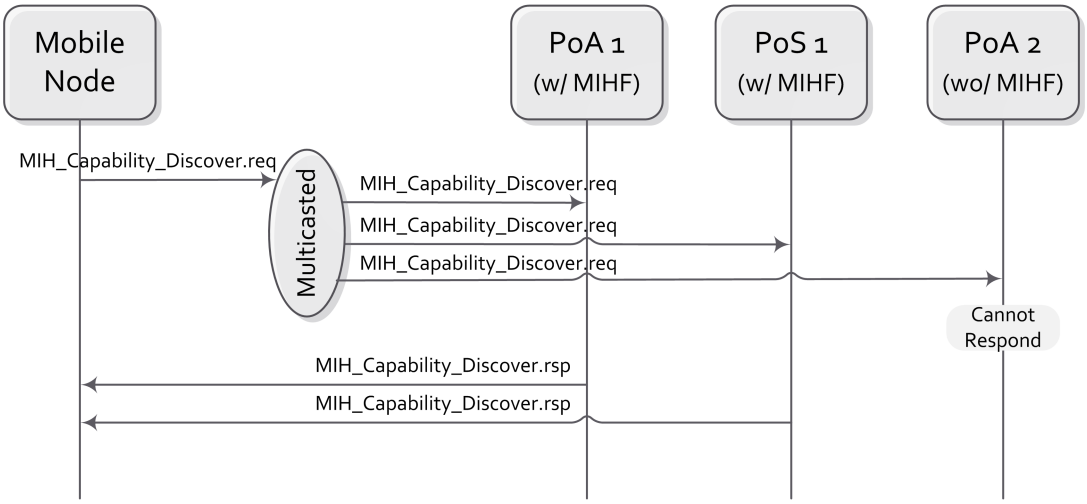

Figure 3.12: MIH Capability Discover procedure

The mechanism depicted is invoked by an upper layer entity, that has just powered up, by using the MIH Services. Since there is no pre-configuration about peer MIHF entities, the MIH User sends a *MIH_Capability_Discover.request* with a multicast MIHF ID. The message will be propagated to all network entities that belong to the multicast group. The MIH-enabled entities will recognize the message and reply with their capabilities with a *MIH_Capability_Discover.response*. Thus, the requestor acquires all the information about the network entities that support the IEEE 802.21 protocol and their capabilities in the coverage area.

3.4 Application of 802.21 to IP Mobility Procedures

IP mobility support is gradually being introduced into network architectures, aiming to provide session continuity support. In the network operator side, the management and control of available resources becomes one of the main concerns. These issues cannot be overcome with actual IP mobility management protocols so they are increasingly being combined with IEEE 802.21. In this way, it is possible to complement IP mobility with mobility services and with an abstract framework that allows upper layers to obtain information and control link layers in an abstract way.

This has been subject of extensive research on several aspects. Seamless mobility over different access technologies using the IEEE 802.21 has been extensively described and evaluated [36] [37] [38] [39], as well as the integration with different mobility management

protocols such as MIPv6 [40] [41] and PMIPv6 [4] [42]. An enhanced MIH framework in a heterogeneous environment, regarding QoS provision in vertical mobility, is presented in [43]. Other studies [4] [44] present a new entity responsible for the mobility decisions and controlling the mobility process.

3.5 Summary

This chapter provided an overview of the IEEE 802.21 standard, focusing on the services and mechanisms available, and it also described the MIH protocol and how peer entities can communicate. In the end, it shows an example of how to integrate MIH services and mechanisms with the MIH protocol in order to discover MIH entities.

In the next chapter are presented the proposed mechanisms to discover MIH entities and their capabilities.

Chapter 4

MIH Discovery Mechanisms

Mobility services (MoS) allow enhanced performance and usability in handover operations by making available a variety of different information types to the MN from different entities within the network and vice-versa. The information exchanged can be of different nature, such as information about the network, commands to perform actions or events about changes in the link conditions [45]. Thus, IP mobility mechanisms described in section 2.2.1 can be complemented with MoS for assisted execution of mobility processes. Due to the dynamic environment induced by mobility, the MN can lack network information for MoS or need to change the MoS for some reason. So, it is necessary to define mechanisms that allow the MN to automatically discover these entities.

IEEE 802.21 defines the PoS as the entity that provides MoS. As mentioned in the previous chapter, the services provided by the PoS are the MIH services and therefore, three distinct service types can be identified: MIIS, MICS and MIES. Whenever a MN needs to interface with a PoS (for registering and using 802.21 features or to give access to MoS) it needs to discover PoS entities in its surroundings, as well as supported IEEE 802.21 capabilities.

Throughout this chapter, the existing PoS discovery mechanisms are discussed for remote scenarios, as well as presenting two novel link layer and MIH-User discovery procedures.

4.1 Local Discovery Mechanisms

Due to the decoupled architecture of the MIHF and the local entities (i.e., Link SAPs and MIH-Users), it is necessary to configure these entities in the MIHF in order to know about its existence. They can be configured statically, by the user, or automatically, using a discovery mechanism. The static configuration can be very limited, and may fail when, for example, the MN activates/deactivates an interface or even hot-plugs new ones. However, the IEEE 802.21 standard does not define any mechanism to discover the local entities so, the development of a novel local discovery procedure is required, allowing the MIHF to discover its Link SAPs and MIH-Users and their capabilities in a dynamic way. Thus, local discovery refers to the procedure that allows the MIHF to discover its Link SAPs and their capabilities, as well as its MIH-Users.

4.1.1 Link SAP discovery

The local discovery mechanisms proposed on this work enables the discovery and management of the available Link SAPs. Thus, Link SAPs discovery (Figure 4.1) is divided

in three main phases: the registration, SAP availability and SAP unavailability.

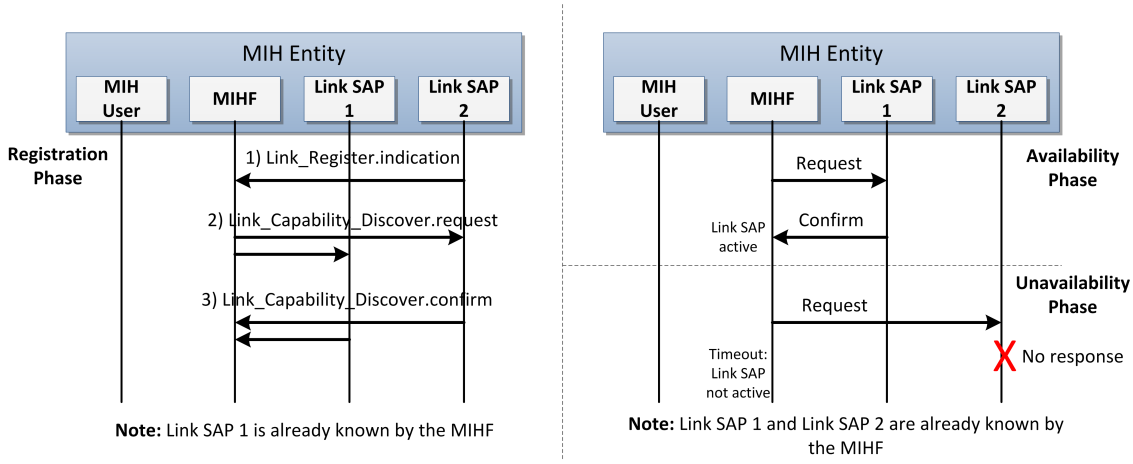

Figure 4.1: Link SAP discovery

In the registration phase (Figure 4.1 - Registration Phase), the MIHF is permanently listening for incoming messages from the Link SAPs at a fixed port. When a Link SAP becomes active, it locally sends a *Link_Register.indication* message (further details about this message are presented in the next chapter) to the MIHF indicating its intention of registering with the MIHF (step 1). This message carries information about the Link SAP, including its ID, listening port, link layer technology and link address. Next, the MIHF requests the Link SAP for its capabilities by sending a *Link_Capability_Discover.request* message (step 2). This process is repeated for all Link SAPs in order to provide the MIHF with the full capabilities belonging to that node.

The second phase is availability (Figure 4.1 - Availability Phase) and, as the name implies, it deals with the Link SAP availability, i.e., the MIHF must be able to detect the active presence of the Link SAP. So, when the MIHF receives a response or an event notification from the Link SAP, it detects the Link SAP as active.

Finally, the third phase deals with SAP unavailability (Figure 4.1 - Unavailability Phase). When the MIHF does not receive a response from a Link SAP, the Link SAP must be considered as inactive or unresponsive. The MIHF must, therefore, initiate the procedures to update its local capabilities.

4.1.2 MIH-User discovery

The process of discovering MIH-Users presented in this work (Figure 4.2) is based on a registration process of the MIH-User with the MIHF. When the MIH-User becomes active, it locally sends a *User_Register.indication* message (further details about this message are presented in the next chapter) to the MIHF, including information about its listening port and function within the mobility process, i.e., if it is a mobility decision entity or not. In this work, the assumption that only one mobility entity can exist at each time is taken in account, so the last MIH-User to register with the mobility decision function will be responsible for it. Upon the receiving of the *User_Register.indication* message, the MIHF learns about the MIH-User and the communications between them can start normally.

The communications between the MIHF and MIH-Users are initiated by the MIH-User,

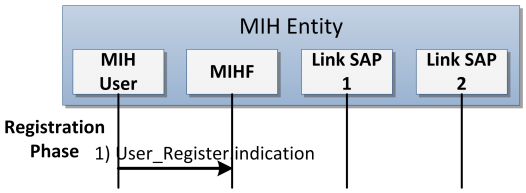

Figure 4.2: MIH-User discovery

in case of request/response messages, or by the MIHF, in case of event notifications messages. As such, and in contrast to the Link SAP discovery mechanism, procedures to detect if the MIH-User is active or not are not defined.

4.2 Remote Discovery Mechanisms

The remote discovery mechanisms involve the discovery of peer MIHFs and its capabilities. As described in the chapter 3.3.4, this is known as MIHF discovery and it can be achieved by using L2 or L3 discovery mechanisms. It allows a MN to discover 802.21-enabled network entities and vice-versa, as well as a network entity to discover each other. In addition, by combining it with the MIH Capability Discover procedure, it is possible to find out the capabilities of the discovered entities, i.e., the services provided by them.

For example, the L2 discovery mechanisms allow the MN to identify which PoAs support MIH mechanisms, helping in the best candidate selection, while the L3 discovery mechanism can be used to discover the PoS that will support the MN mobility in the current access network.

4.2.1 L2 Discovery Mechanisms

Peer MIHF discovery at L2 is performed either in a media specific manner (i.e., listening to enhanced media dependent broadcast messages) or by using multicast data frames (i.e., L2 Capability Discover Exchange). These approaches are technology dependent and, therefore, only enable the discovery of MIHF entities that support the same L2 technology. For example, a MN making use of its Wi-Fi interface can only listen for L2 IEEE 802.11 frames so, it can only discover PoA entities that are broadcasting messages through the same technology. In addition, these mechanisms are more suitable to discover entities where the PoS is co-located with a PoA, since the MN must be able to communicate with them via L2 Management frames.

For IEEE 802.11 and IEEE 802.16, the MIH protocol messages can be sent before the authentication over the management plane by using respective media specific MAC management frames [7]. In this work it is only referred the IEEE 802.11 possibilities and the following two approaches (Figure 4.3) were identified:

- **Listening to the Enhanced Media Dependent Beacons:** L2 Beacon frames are enhanced with node's capabilities, by encapsulating an *MIH_Capability_Discover.response* message within the existing IEs of the frame (step 1). The MN, by scanning the media, can identify the available PoAs and which MIH services they support. This is also known as unsolicited capability discover, since no prior *MIH_Capability_Discover.request*

has been made, and therefore forwarding the results to the MIH-User is not mandatory (step 2).

- **L2 Capability Discover Exchange:** this approach considers the discovery of PoA/PoS entities and their capabilities in one single transaction. To achieve this, the MN enhances the information sent in a L2 Probe Request frame with its own capabilities (step 2), by introducing a *MIH_Capability_Discover.request* message within the existing IE. The PoA, receiving this message, encapsulates its own capabilities in a L2 Probe Response frame and sends it to the MN (step 4).

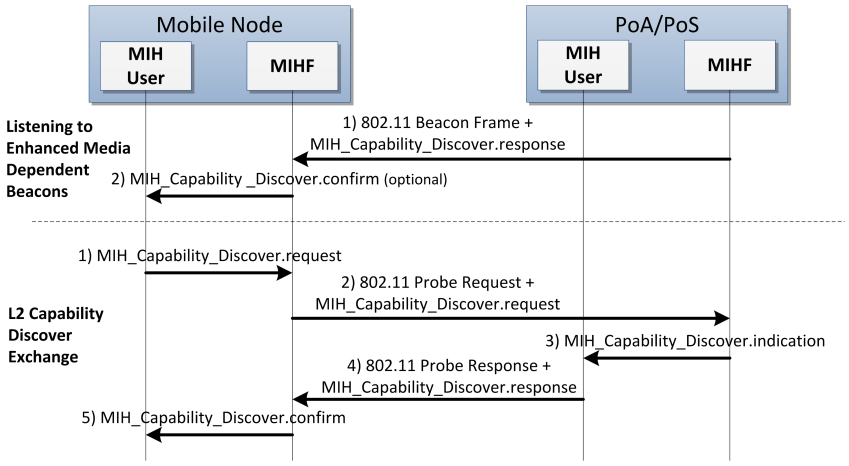

Figure 4.3: MIH services discovery using L2 mechanisms

4.2.2 L3 Discovery Mechanisms

L3 mechanisms are mainly used to obtain the transport information needed for remote communications using the MIH Protocol. Such information includes the transport address and the listening port of the peer MIHF entities, as well as its MIHF ID and which MIH services these entities provide. MIHF entities can be located anywhere (i.e., in the home or visited network or even in a third-party network), creating a variety of deploying possibilities based on the location of the PoS and the MN (described in chapter 3.2.4.5). By using L2 discovery mechanisms, it is only possible to discover PoS that are in the node's L2 range. Thus, L3 discovery mechanisms becomes a key part of a broader discovery process, since they are not restricted to physical limitations. By using L3 mechanisms, it is possible to discover the PoS independently of the location of the MN, and it also enables the discovery of the PoS whether or not it is co-located with PoAs.

The discovery of the PoS via L3 mechanisms can be achieved by using the MIH Capability Discover procedure defined in the IEEE 802.21 standard [7] and/or the mechanisms defined in [30] that involve using DHCP [35] and DNS [34] services. Thus, this work proposes an enhanced architecture to support these discovery mechanisms. Figure 4.4 depicts the proposed architecture and its component location within a node.

Compared with the standard architecture of the IEEE 802.21, the node has two new components: DHCP and/or DNS users. These are seen as MIH-enabled entities of the upper layers (i.e., MIH-Users) with which the MIHF interacts to request the discovery of available

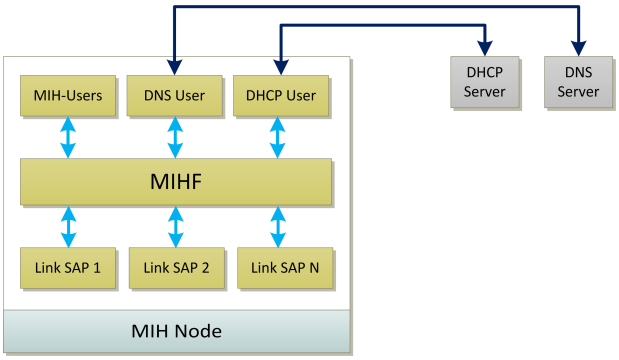

Figure 4.4: L3 Discovery Mechanisms Architecture

PoS. The DNS and the DHCP clients are then responsible for directly communicate with the respective server, in order to discover PoS entities and to obtain associated transport information.

4.2.2.1 Using Unicast/Multicast MIH Capability Discover

The IEEE 802.21 standard only provides mechanisms to discover the capabilities of the MIHF entities, although it can be used to discover peer MIHF entities too, by multicasting the messages. A *MIH_Capability_Discover.request* message can be sent in a unicast way to a specific MIHF destination or in a multicast way to a multicast domain. Unlike the unicast case in which the MIHF destination is already known, in the multicast case the requestor MIHF can receive responses from unknown MIHF entities. The MIHF can learn not only the MIHFs capabilities but also the transport information of each MIHF. However, this can only be used in scenarios where the PoS and the MN are located in the same network domain, unless a subscription to the multicast domain has been made.

Figure 4.5 presents an example of a broadcast MIH Capability Discover initiated by the MN. Initially, the MN sends a *MIH_Capability_Discover.request* message with a multicast MIHF ID destination to a broadcast domain (steps 1 and 2). By receiving this message, all MIH-entities able to respond to multicast messages will transmit their capabilities to the MN, by sending a *MIH_Capability_Discover.response* message in a unicast way (step 4). Finally, the results are forwarded to the MIH-User and the discovery process has ended (step 5).

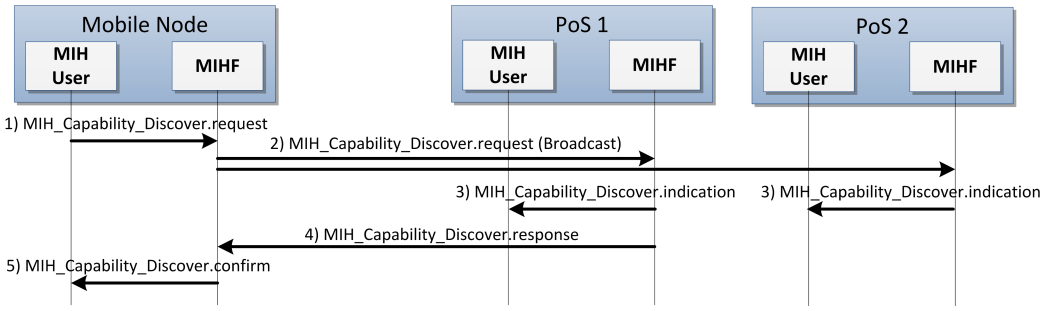

Figure 4.5: Broadcasted MIH Capability Discover message

When this process ends, the MN has enough information to register with the discovered MIHFs (in this case, only PoS 1), initiating its remote interaction by sending commands,

registering events or requesting information about the network.

4.2.2.2 Using DHCP

[35] defines how the discovery of PoS can be achieved using DHCP services. It introduces additional DHCP options (Figure 4.6) which include a list of IP addresses and/or a list of domain names that can be mapped to servers providing IEEE 802.21 services.

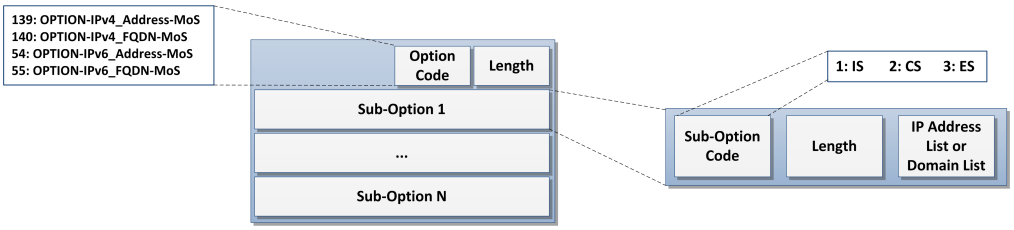

Figure 4.6: Specific 802.21 DHCP Options

When a MN lacks the network information for accessing MIH services, or when it needs to change the MIH service (e.g., due to handover or to recover from the single point of failure of the current PoS), it must initiate the procedures to discover a new PoS. Figure 4.7 depicts the discovery procedure using DHCP services, which was enhanced with an optional MIH Capability Discover exchange, allowing the PoS and the MN to become aware of the each other capabilities.

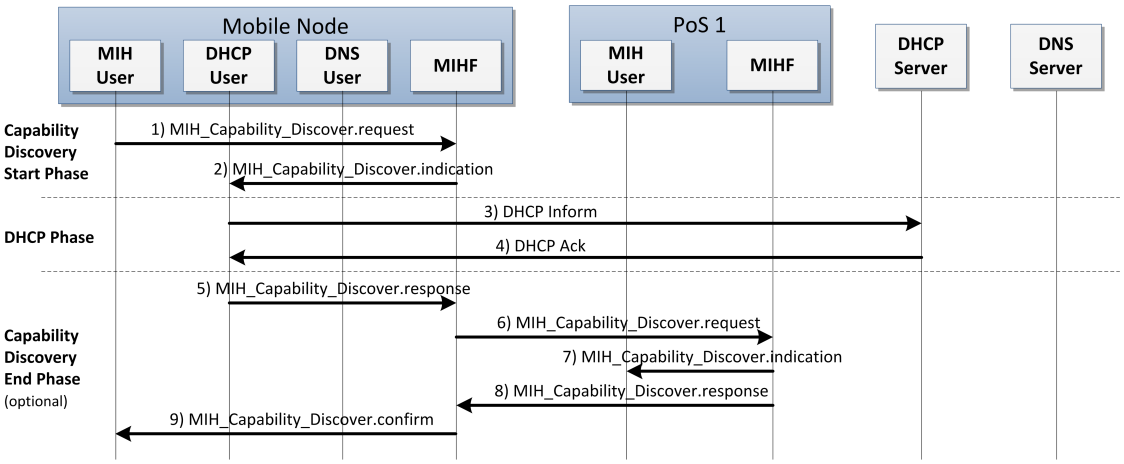

Figure 4.7: PoS discovery using DHCP

The MIH-User, aware that it needs to discover a new PoS, initiates the discovery procedure by sending a *MIH_Capability_Discover.request* message to a multicast MIHF ID destination (step 1). Depending on the deployed configurations, a MIH-enable DHCP client, triggered by a *MIH_Capability_Discover_indication* (step 2), includes the specific 802.21 options (in this case a *MoS IP Address Option*) in the Parameter Request List (PRL) of the respective DHCP message, querying a DHCP server to obtain the IP address of the PoS and which services it provides (steps 3 and 4). At this point, the MIHF has the transport information of the discovered PoS and it can initiate the MIH Capability Discover procedure (steps 5 to 9).

At this point, the MN is not requesting for an IP address for itself and therefore, only DHCP Inform and Acknowledge messages are used.

4.2.2.3 Using DHCP at bootstrap

In the previous scenario, it was considered that the MN was already connected to the network. Nevertheless, the discovery procedure can also be done at bootstrap (Figure 4.8), allowing the MN to request the required information for PoS configuration during the initial address configuration, after a successful L2 network attachment. The MIH-enable DHCP client triggered by the L2 attachment event notification (step 1), initiates the DHCP procedures to request an IP address for the MN itself (steps 2 to 5). Thus, and like the previous case, it includes the specific 802.21 options (in this case a *MoS IP Address Option*) in the exchanged messages, such as DHCP Discover, Offer, Request and Acknowledge messages.

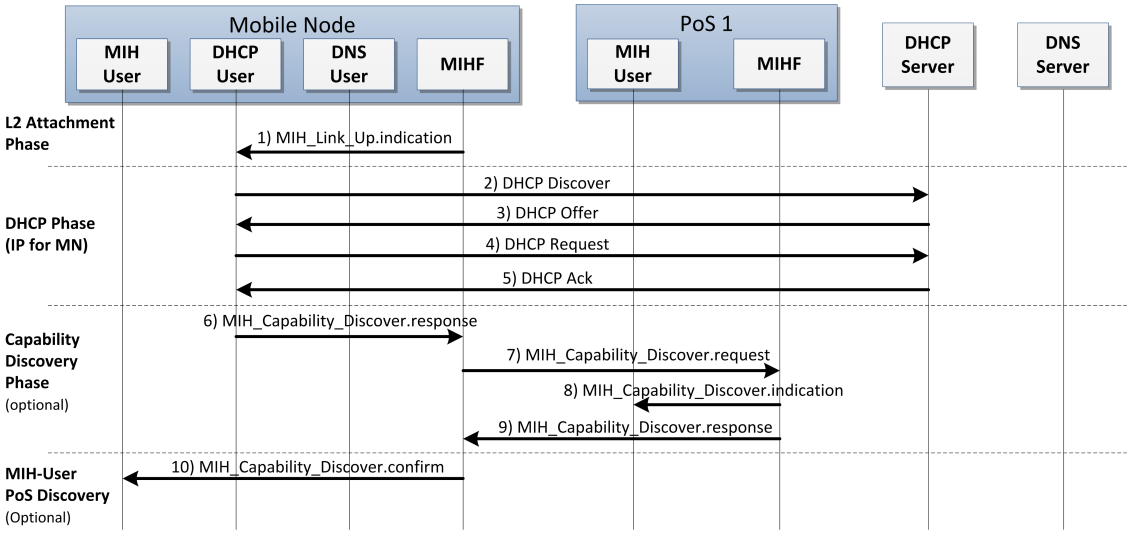

Figure 4.8: PoS discovery using DHCP at bootstrap

Since the discovery process was not triggered by a MIH-User, it is classified as an unsolicited discovery and, therefore, the enhancement of this procedure with the additional MIH Capability Discover exchange is not mandatory (steps 6 to 9), as well as the forwarding of the PoS capabilities towards the MIH-User (step 10). However, if the PoS capabilities are forwarded to the MIH-Users, it must be sent to all of them, since there is no specific requestor.

4.2.2.4 Using DNS

[34] defines DNS procedures (Figure 4.9) that enable the discovery of PoS entities within a given domain. It allows the discovery of PoS information, such as IP address, listening port, transport protocol, MIHF ID, as well as the services provided by it. Like in the DHCP scenario, the discovery entity (i.e., the MIH-enable DNS client) is triggered by a *MIH_Capability_Discover.indication* (step 2). By performing a Naming Authority Pointer (NAPTR) query for the configured domain, it receives the records which identify the available MIH services and respective transport protocols in the requested domain (steps 3 and 4). The DNS-User is then able to request a service location query, enabling the discover of the IP address and the listening port of the PoS (steps 5 and 6).

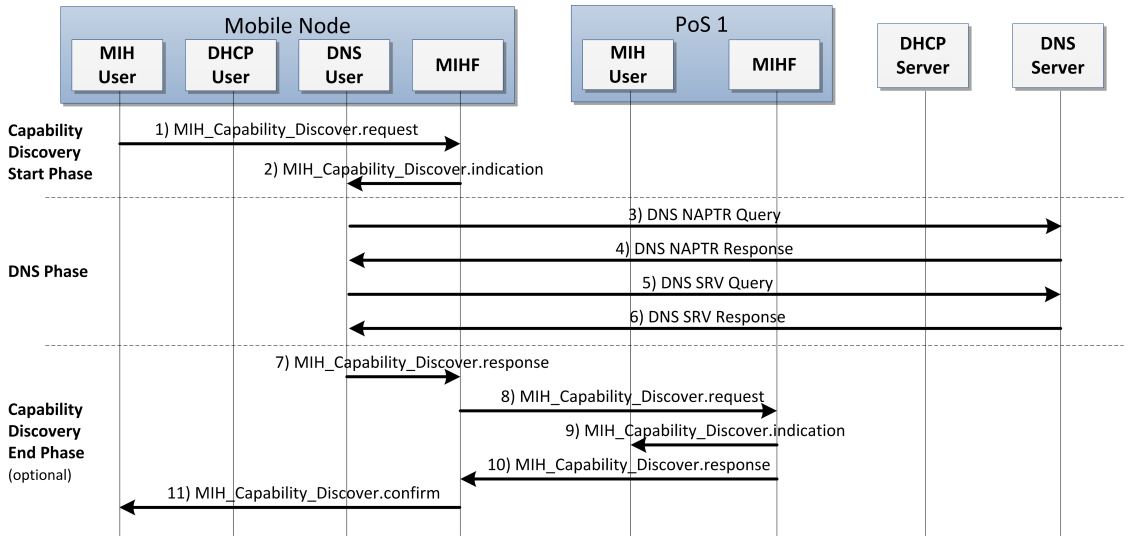

Figure 4.9: PoS discovery using DNS

The MIH Capability Discover exchange between the MN and the PoS, described in the previous scenarios, is also applicable here (steps 7 to 11).

4.2.2.5 Using DHCP and DNS

The isolated use of each of these mechanisms is limited in terms of supported scenarios. Therefore, the combination of different discovery mechanisms enables the discovery of PoS in several different scenarios, which could not be possible by using a single mechanism. This work presents an example of this combination, in which the integration of the DHCP and DNS mechanisms enable a MN to discover PoS entities located on different domains, i.e., when the MN is in a visited network and the PoS is in the home network or when the PoS is within a third-party remote network.

This scenario describes an integration of DHCP and DNS procedures, presented in Figure 4.7 and Figure 4.9 respectively. Depending on the deployment configurations, when the discovery of the PoS is done via DHCP (by introducing a specific 802.21 *MoS FQDN Option*), the DHCP server, instead of replying with an IP address, can reply with a domain name list. With this information the MN cannot know which PoS are available, so it must use a complementary mechanism, like DNS, to discover the PoS located in the received domains. The signalling of this procedure (Figure 4.10) consists on a combination of the DHCP (steps 3 and 4) and DNS phases (steps 7 to 10), with the DHCP procedure conclusion triggering the DNS phase (steps 5 and 6). When the DNS phase ends, the MIHF already has all needed information to communicate with the discovered PoS and hence it initiates the capabilities exchanged between the MN and the discovered PoS (steps 11 to 15), which is similar to the one described in the previous scenarios. In this scenario, the discovery process is initiated by the MIH-User (step 1) and the bi-directional capabilities exchange between the MN and the PoS is optional.

Notwithstanding, this joint usage of DHCP and DNS procedures can also be done at bootstrap (Figure 4.11). The initial procedure is identical to the one described in Figure 4.8, i.e., the discovery procedure is triggered by a L2 attachment event notification (step 1).

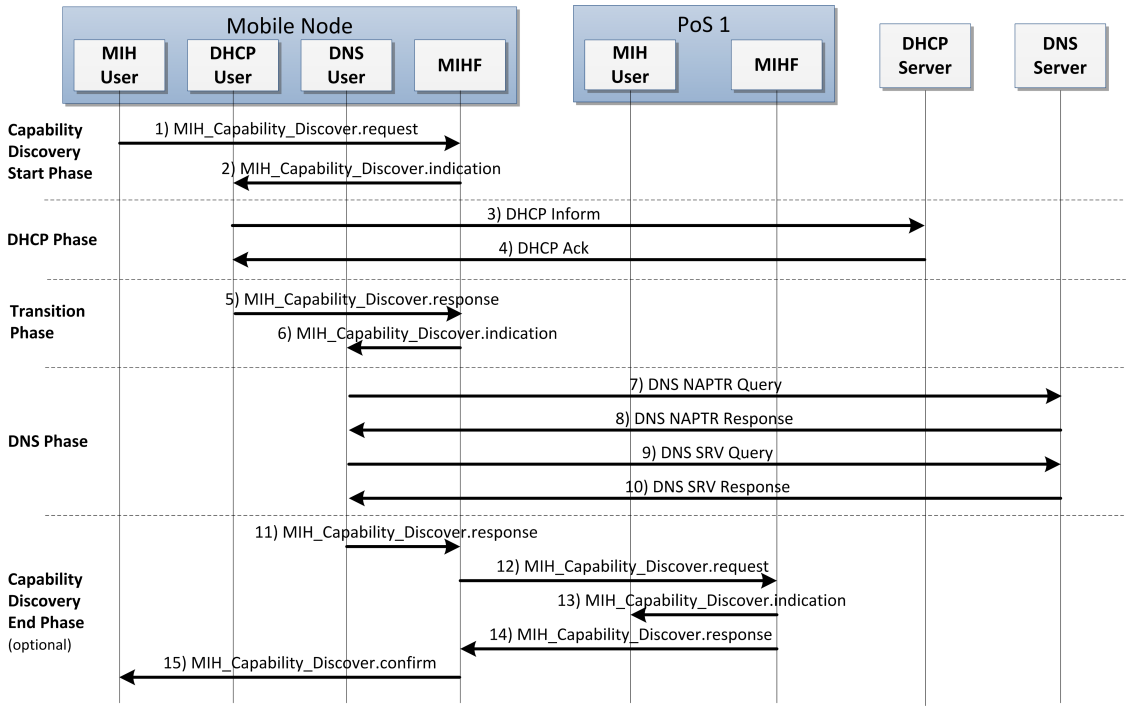

Figure 4.10: PoS discovery using DHCP and DNS

Then, the DHCP-User initiates its procedures to acquire an IP address and, at the same time, requests for PoS information (steps 2 to 5). The conclusion of the DHCP phase triggers the DNS phase (steps 8 to 11) in order to discover the remaining PoS information. Since, this is an unsolicited discover, the MIH Capabilities exchange (steps 12 to 15), as well as the forward of the capabilities to the MIH-Users (step 16), are not mandatory.

4.3 Summary

This chapter described the existing discovery mechanisms for remote scenarios, as well as the introducing of a new local discovery procedure for Link SAPs and MIH-Users. It can be accomplished over L2 or L3 and can be used isolated or through the combination of several discovery mechanisms, in a complement way.

In the following chapter, the implementation of these mechanisms made over an open-source 802.21 implementation is discussed.

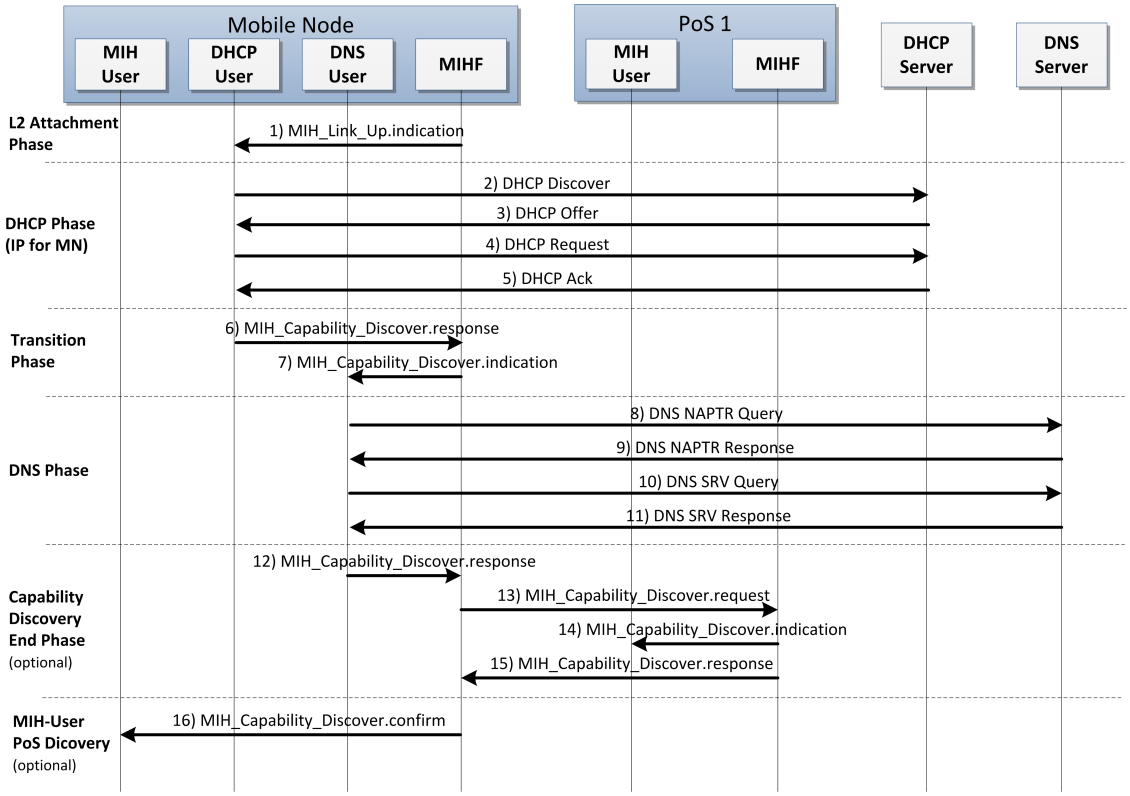

Figure 4.11: PoS discovery using DHCP and DNS at the bootstrap

Chapter 5

Implementation of MIH Discovery Mechanisms

The MIH discovery mechanisms presented in the previous chapter were developed over an open-source 802.21 implementation named ODTONE (see Annex A), and thus constitute evolutions and adaptions of these mechanisms over the base IEEE 802.21 standard. This chapter describes the key elements of the implementation of these mechanisms, in what concerns the changes in the MIHF and the MIH entities created. The additional features developed in the ODTONE framework are also described in this chapter.

5.1 Local Discovery Implementation

One of the requirements for the implementation of the local discovery mechanisms was the support for multiple Link SAPs and MIH-Users. Initially in ODTONE, there was a unique component used to store the information about MIH entities (MIH-Users, Link SAPs and peer MIHFs). To provide a better management of the information of each entity, this component was divided in three distinct ones (Table 5.1), each one used to store the information about each MIH entity. This extension done over the ODTONE software allows a more efficient way for the MIHF to find the Link SAP or MIH-User to whom it must redirect the received messages. For example, the messages from the MIH-Users or peer MIHFs, whose destination is a Link SAP, include link layer information, such as link type and link address. The MIHF, making use of that information, searches in the *Link_book* for the correspondent Link SAP and then forwards the message. The same behaviour occurs for messages destined to the MIH-Users, however the MIHF uses subscription information or information about the mobility decision entity to redirect the messages.

For both Link SAPs and MIH-Users discoveries the IEEE 802.21 standard does not define any MIH message, primitive or TLV that fit the requirements so it was necessary to create and implement them as extensions to the ODTONE software. Figure 5.1 depicts the created *Link_Register.indication* and *User_Register.indication* messages.

The *Link_Register.indication* message contains, in addition to the mandatory source and destination TLVs, a newly created TLV, named *Interface Type Address TLV*, which carries the network interface information. The same happens with the *User_Register.indication* message, although with a newly created TLV, named *Handover Support TLV*, that carries a boolean value indicating if the MIH-User is a mobility decision entity or not.

Table 5.1: MIH entities storage structures

Component	Stored Information	Description
Link_book	IP address, port, link identifier (technology type and link address) and number of consecutive fails.	Used to store the information about all known Link SAPs. It makes the correspondence between the Link SAP MIH ID and the information stored.
User_book	IP address, port and mobility function.	Used to store the information about all known MIH-Users. It makes the correspondence between the MIH-User MIH ID and the information stored.
Address_book	IP address, port and supported transport.	Used to store the information about all known peer MIHFs. It makes the correspondence between the MIHF ID and the information stored.

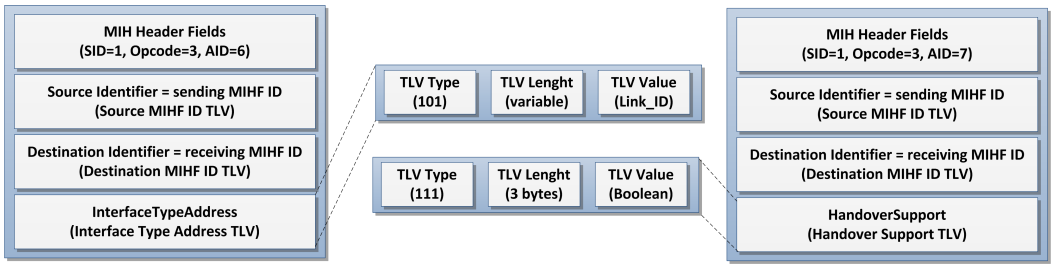

Figure 5.1: *Link_Register.indication* and *User_Register.indication* messages

5.1.1 Link SAP discovery

When a Link SAP wants to dynamically attach to a MIHF, it initiates the registration procedure described in the previous chapter and, therefore, the MIHF must be able to recognize this procedure to properly handle the discovery process. Thus, after receiving a *Link_Register.indication* message, the MIHF processes it and stores the received information about the Link SAP in the *Link_book*. The network interface information, such as technology type and link address, is extracted from the *Interface Type Address TLV*, while the IP address and port are discovered based on the network information presented in the received packet. At this point, the Link SAP is attached to the MIHF and, consequently, the MIHF capabilities change. In this way, the MIHF must update them, so it requests all Link SAPs attached to it (i.e., all Link SAPs that exist on the *link_book*) about their capabilities by sending a *Link_Capability_Discover.request* message, merging the results with its current capabilities.

SAP availability and SAP unavailability phases were implemented using specific timers, which were added to the base ODTONE implementation. When the MIHF forwards a request message to a Link SAP it sets a timer (this value is configured by the user) during which it waits for the reply. If the response is received within the time interval, the MIHF acknowledges the presence of the Link SAP, i.e., it detects that the Link SAP is still active. In addition, the presence of the Link SAP can be detected when the MIHF receives a link event notification from the Link SAP. Thus, the MIHF resets the respective *fail* field in the *Link_book*. If the response is not received until the timer expires, it could mean that the Link SAP is not available anymore (i.e., the respective interface was shut down by the user) or is unresponsive

and, therefore, the MIHF may forget the Link SAP.

There are several reasons why the response is not received within the time interval, such as delays, packet lost, unsupported request, among other reasons. As such, there must be a confirmation that the Link SAP is really down before its removal. If the MIHF does not receive the response message, it registers it as a failure response and when the number of consecutive failure responses reaches a defined threshold (this value is configured by the user) the Link SAP is removed and the MIHF local capabilities are updated.

It may happen that the MIHF forwards the request message to multiple Link SAPs. In these cases, the timer has a double function: to check the availability of Link SAPs and to wait for all responses before the MIHF replies to the requestor. Then, after the timer expires, the associated handler is called, which processes the received messages in one unique message and forwards it to the requestor. Since the messages are not processed as they arrive to the MIHF it is necessary to store them. For this purpose, the pending messages are grouped, by its TID, in a new component named *link_response_pool*.

5.1.2 MIH-Users discovery

In order to avoid simultaneous decisions on the handover, the existence of a single mobility decision entity was defined, to which it will be forwarded all messages related to the handover procedure. This actually reflects the typical mobility management entity deployed where a single controller module (either in the MN, or the network or both acting in synchrony) are in chase of the handover control processes.

When the MIHF receives the *User_Register.indication* message, it stores the MIH ID, IP address and listening port of the MIH-User and it verifies with which function on the mobility process the MIH-User is registering. If the MIH-User is registering as the mobility decision entity, it becomes the new mobility decision entity replacing the old one and, from now on, all *MIH_***_HO_*** commands received by the MIHF are redirected to it.

5.1.3 Additional features

The development of the local discovery mechanisms led to the implementation of additional features related to the management of MIH-Users and Link SAPs, done over the ODTONE implementation.

5.1.3.1 Link Capabilities Discover Optimization

The additional feature more related to the discovery mechanisms is the possibility of storing the capabilities in the MIHF. Thus, instead of requesting all Link SAPs for the supported events and commands, each time the MIHF receives a *MIH_Capability_Discover.request* message, the MIHF immediately replies with its capabilities. This optimization brings several benefits, such as the reduction of the response time, the number of exchanged messages and the overhead of the MIHF and Link SAPs. The differences between the two procedures are explained in Figure 5.2. Nevertheless, it implies the constant update of the local capabilities, which is supported by the implemented local discovery mechanism. It also allows the storage of peer MIHF capabilities, although there are not yet any mechanism that use this feature.

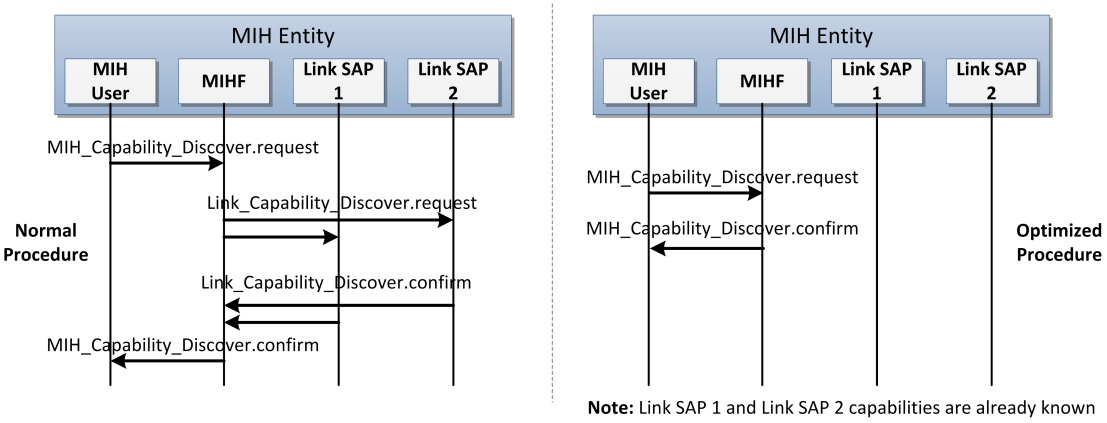

Figure 5.2: Optimization of Link Capabilities Discover

5.1.3.2 Mapping between *MIH_LINK_SAP* primitives and MIH protocol messages

ODTONE uses the MIH protocol to provide communications between the MIHF and local entities. However, it was verified that some *MIH_LINK_SAP* primitives cannot be mapped in the existing MIH protocol messages so, it was necessary to define new TLVs that carry the information presented in these primitives. The Table 5.2 makes the correspondence between the *MIH_LINK_SAP* primitives and the new defined TLVs.

Table 5.2: Mapping between the MIH_LINK_SAP primitives and the new defined TLVs

MIH_LINK_SAP Primitive	**TLV Type Name**	**TLV Type Value**	**Data type**
LinkParametersRequest	tlv_link_parameters_req	102	LIST(LINK_PARAM_TYPE)
LinkParametersStatusList	tlv_link_parameters_status_list	103	LIST(LINK_PARAM_TYPE)
LinkStatesRequest	tlv_link_states_req	104	LINK_STATES_REQ
LinkStatesResponse	tlv_link_states_rsp	105	LIST(LINK_STATES_RSP)
LinkDescriptorsRequest	tlv_link_descriptor_req	106	LINK_DESC_REQ
LinkDescriptorsResponse	tlv_link_descriptor_rsp	107	LIST(LINK_DESC_RSP)
LinkAction	tlv_link_action	108	LINK_ACTION
LinkActionResult	tlv_link_ac_result	109	LINK_AC_RESULT
ScanResponseSet	tlv_link_scan_rsp_list	110	LIST(LINK_SCAN_RSP)
LinkDetInfo	tlv_link_det_info	111	LINK_DET_INFO

5.1.3.3 Event Subscribe/Unsubscribe Optimization

Other additional feature implemented was the optimization of the event subscribe mechanism (Figure 5.3 - A), by reducing the number of exchanged messages. When a MIH-User requests for an event subscription, if a subscription has been made previously by other entity, the MIHF subscribes the MIH-User and immediately replies to it, otherwise the MIHF firstly tries to subscribe the event with the Link SAP or the peer MIHF.

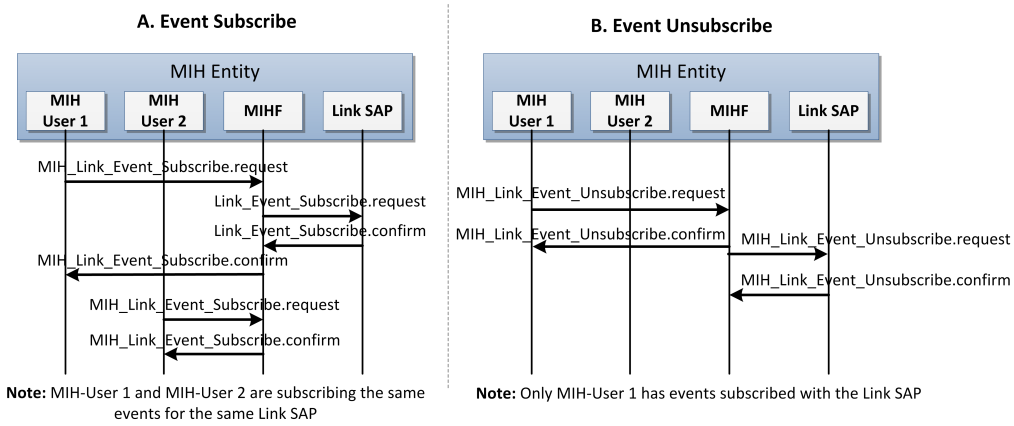

Figure 5.3: Event Subscribe and Event Unsubscribe optimization

The method to optimize the unsubscription mechanism (Figure 5.3 - B) is similar to the subscription one and it aims for the same objectives. When a MIH-User requests to unsubscribe an event in a Link SAP the MIHF unsubscribes it locally. The MIHF unsubscribes the event with the Link SAP or the peer MIHF only if the event has no more subscriptions.

5.1.3.4 Remote Link SAP handling

Since the *Link_book* module allows the storage of the IP address of the Link SAPs, the Link SAPs attached to the MIHF can be on the same machine or not, i.e., the Link SAPs can be local or remote (Figure 5.4).

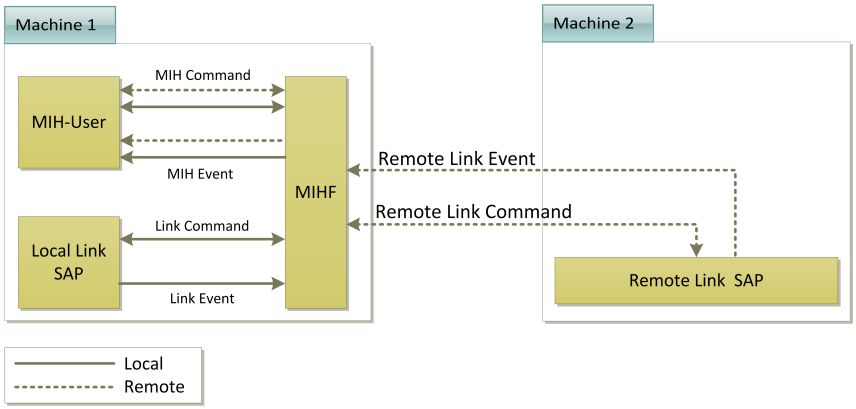

Figure 5.4: Location of the Link SAPs

Although the communications are made between two different machines, the remote Link SAP is always seen as a local entity to the MIHF and therefore, the messages received from local and remote Link SAPs are processed equally. This behaviour is possible because ODTONE uses sockets to provide local communications.

5.2 L2 Discovery Implementation

The management of L2 aspects is highly dependent on the technology and the operation system (OS). Each technology has its own L2 aspects. In addition, the network device management requires certain functions and/or capabilities from the host OS. To overcome this issue, the implementation of the L2 discovery mechanisms followed a modular approach, which enables the coupling of different mechanisms according to the L2 technology and the host OS. In what concerns this dissertation, the implementation focuses on WLAN aspects over the Linux OS.

To provide an abstraction from lower layers and OS details, the designed solution is based on a driver architecture approach, which aims to provide an interface that facilitates the integration of the L2 specific implementations with the ODTONE architecture. Thus, a new management entity was specified, over the ODTONE implementation, embedded in the *service management* module (Figure 5.5), named *discovery service*, that allows the management of the following aspects:

- **Initializations:** manages the initialization of the L2 Controllers responsible for each network interface.
- **Scanning:** manages the periodic scan triggering of the media.
- **Broadcasting:** manages the periodic update of the MIH Capability Discover message sent in the L2 specific messages. This message consists of a MIH Capability Discover response message with a multicast MIHF ID destination.

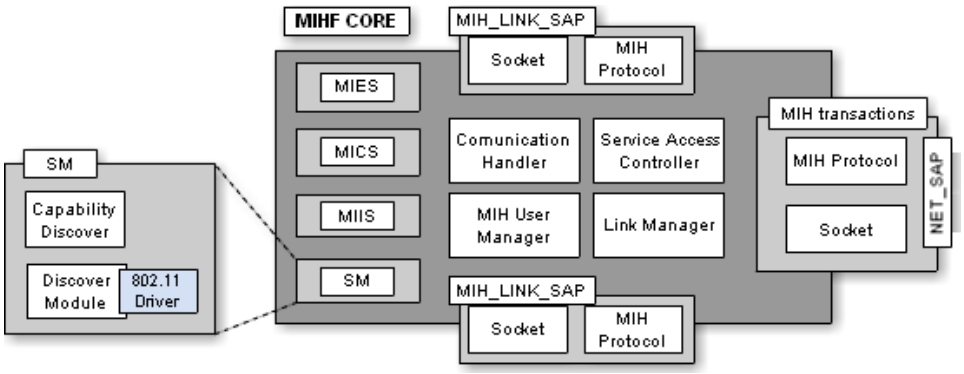

Figure 5.5: Integration of the discovery service module in the ODTONE architecture

The *discovery service* module is just a management entity and does not define any L2 specific behaviour from lower layers. In order to control the aspects related to the IEEE 802.11 technology, a driver that uses the Netlink Protocol Library Suite (*libnl*) [46] together with the *nl80211* [47] was developed. The first is a collection of libraries providing Application Programming Interfaces (APIs) to the *netlink* protocol based in the Linux kernel interfaces, while the second is a header that defines the available system calls to interact with *cfg80211*.

For each IEEE 802.11 network interface an instance of this driver must be created, which works not only as a listener for the L2 802.11 frames but also as an entity to control and to manage the respective interface. This driver has three main operations: initialization, scanning and encapsulation.

In the initialization, a subscription for the MLME and SCAN multicast groups is made, which enables the forwarding of messages belonging to these groups to the user space. A registration to receive Probe Requests, Probe Responses and Action Management Frames was also made. At the end of the initialization operation, the procedures for scanning and broadcasting in the media can be triggered. The scan operation allows the discovery of new MIH-enable PoAs, by triggering scans in the media and forwarding the results to a given handler. This handler is then responsible for verifying if an encapsulated MIH message is presented (by parsing the IEs contained in the received messages, looking for IEs with the *Type Value* equal to 51), forwarding it to the MIHF's core. Finally, the encapsulation operation defines a way to encapsulate MIH messages in an IE of the Beacon Frames. Since the IEEE 802.21 standard allows the exchange of information in an unauthenticated state using IEEE 802.11 Management Frames, the ability to send these frames with the encapsulated MIH messages in its IEs was also implemented. These frames can be sent to a specific destination or can be broadcasted.

In addition, to support communications over L2 and the co-existence with the L3 communications mechanisms, the transport information was added to the metadata of a MIH message. Thus, the MIHF labels all MIH messages received with information about its transport (Table 5.3), allowing the MIHF to reply using the same transport mechanisms.

Table 5.3: MIH messages metadata

Transport Mechanism	**Information**
L2	L2 Technology L2 Address Frequency Network interface controller
L3	IP address Port

5.3 L3 Discovery Implementation

To provide communications between the discovery entities and the MIHF in an independent way (i.e., with no dependencies on the API provided by the discovery entities), the discovery entities were implemented as MIH-enabled entities, allowing the message exchange to be done using the MIH protocol.

However, the MIH protocol does not support the necessary mechanisms to carry the information about the discovered entities. Therefore, a new optional TLV type was created and added to the MIH Capability Discover messages of the base ODTONE implementation. Consequently, a new data type was defined as well. Figure 5.6 represents the TLV type and the data type created.

The *mos_dscv* data type carries the discovered PoS divided by service type. Each PoS is identified by an instance of the *mos_info* data type, which carries information about its ID, IP address and listening port.

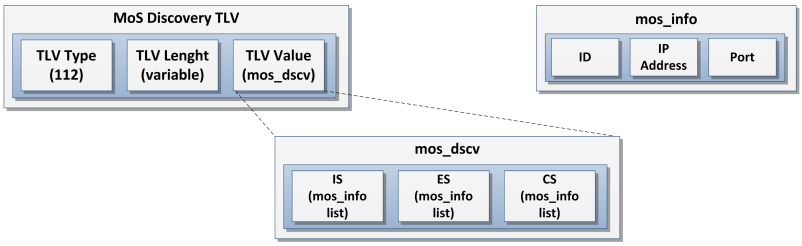

Figure 5.6: TLV type and data type created and its integration on the MIH Capability Discover message

5.3.1 MIHF

The management of the discovery signalling, as well as the procedures depicted below, were implemented over the *discovery service* module, presented in the L2 implementation details.

The L3 discovery mechanisms can be enabled by defining a ordered list of discovery entities in the MIHF configuration file. This list also defines the order by which the entities will be requested for discovering PoS. For example, if no entity is configured with discovery function, the discovery mechanisms will not be enable and therefore, the discovery mechanisms are the default ones, i.e., it is only possible to discover entities by multicasting MIH Capability Discover messages as defined in the IEEE 802.21 standard. Otherwise, the discovery mechanisms are enabled and, upon the receiving of a discovery request, the MIHF invokes the first discovery entity configured. By receiving the discovery results, the MIHF checks if all necessary information about the PoS was discovered and, based on that, decides to complement it with other discover mechanism or to initiate the MIH Capability exchange procedure. The complete behaviour of the MIHF can be seen in the Figure 5.7.

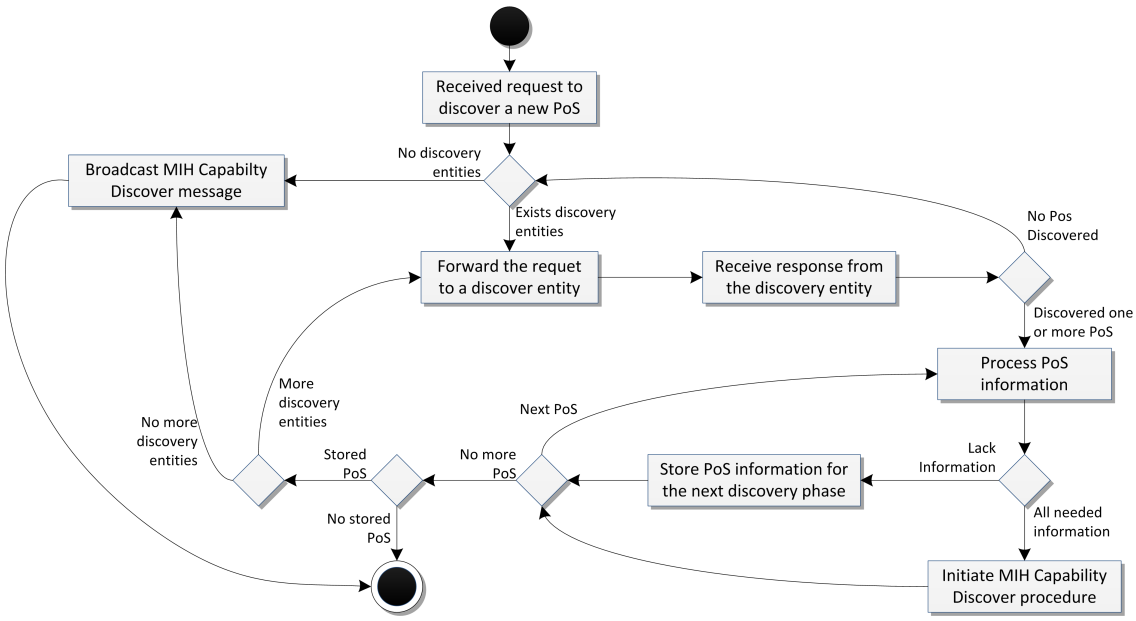

Figure 5.7: MIHF Activity Diagram

The discovery entities retrieve the discovered information to the MIHF divided by services,

i.e., for each service the MIHF receives a list of available PoS, which are individually requested for its capabilities. However, with this approach, if a given PoS provides more than one service it appears in more than one list and therefore, it is requested more than once for its capabilities. To avoid this repetition the MIHF tracks the requested PoS and, if it detects a repeated one it drops the request.

Usually, it is the MIH-User that requests for a new PoS discovery by sending a *MIH_Capability_Discover.request* to a multicast MIHF ID destination. Nevertheless, the discovery process can be autonomously initiated by the discovery entity and, in these cases, the MIHF, if configured, forwards the results to all MIH-Users registered in the *User_book* (except the discovery entities).

5.3.2 DHCP-User

The implementation of the DHCP-User was made as well over ODTONE and its operation focuses on the L3 discovery mechanisms described in the previous chapter (such as messages, interactions with other entities and signalling). It is based on an already existent open-source implementation of the DHCP client. The *"dhclient"* and the *"dhcpcd"* were the two options on the table. To evaluate which implementation best fits the necessary requirements, several tests to each one were done, such as management of several interfaces, support for IPv4 and IPv6 and documentation. Since the *"dhclient"* had better results on the management of several interfaces (one of the main selection parameters), it was the chosen one to be modified in order to become MIH-enabled. Hence, the DHCP-User is a modification of the *"dhclient"*, which was extended with the capability to communicate using the MIH protocol.

Its class diagram, which is presented in the Figure 5.8, has the following components:

- **DHCP User:** this component is responsible for the configuration of the network interfaces, as well as the registration with the MIHF and the management of the MIH signalling.
- **Log:** this component is defined as a singleton and it provides logging capabilities to the others components.
- ***dhclient* subsystem:** this represents the *dhclient* core, which was converted in a set of functions that allows the control of DHCP procedures.
- **mih:** this represents a set of functions that facilitates the *dhclient* core to handle the 802.21 elements.

The DHCP-User must be configured with the information about the network interfaces that it will manage. Thus, when it starts running, it can initiate the procedures to configure these interfaces, in order to correctly control them. One of the initial procedures is to subscribe the 802.21 *MIH_Link_Up* and *MIH_Link_Down* events, for each interface. This enables the DHCP-User to be notified about L2 attachments or detachments, allowing it to make a better management of the interfaces, as well as to properly initiate the DHCP mechanisms.

The MIHF triggers the discovery process in the DHCP-User by sending a MIH Capability Discover indication message. In this case, the DHCP-User can request the desired information, by sending a DHCP Inform to all up interfaces (an interface is considered up if previously received a notification about an L2 attachment). The discovery process can also be initiated by the detection of a L2 attachment, i.e., by receiving a *MIH_Link_Up.indication* message. This

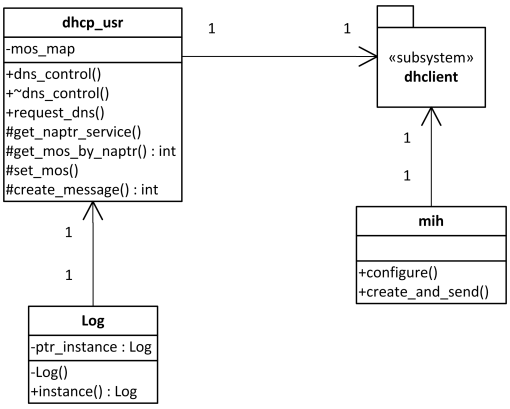

Figure 5.8: DHCP-User class diagram

trigger notifies the DHCP-User to initiate the normal procedures for getting an IP address, which is enhanced to request the PoS information. Due to the DHCP renewal procedure, it is also possible to periodically receive information about the available PoS. The information about the discovered PoS is joined together in one single *MIH_Capability_Discover.response* message and sent to the MIHF.

However, it was necessary to implement the DHCP INFORM message, which affected the *"dhclient"* state machine (Figure 5.9) with the addition of a new state (*"S_INFORM"*). This new state was essential to correctly detect the response of the DHCP INFORM message and to avoid the IP assign procedures.

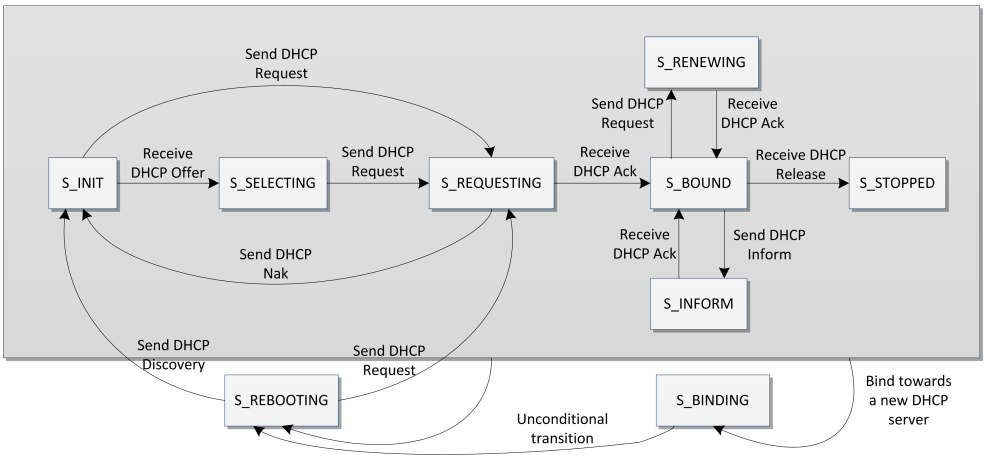

Figure 5.9: Modifications on the *"dhclient"* state machine

Although the DHCP library provides support for defining new DHCP Options through the configuration file of DHCP client and server, it was extended to support 802.21 specific DHCP Options. In this way, the configuration of these options became easier and avoids possible configuration errors.

Each DHCP option has a name, a code and a structure. The name is just an identifier to refer to an option, the code is a number, used by the DHCP client and server, to refer to an option and, finally, the structure represents the data type carried by the option.

The structure of an option is simply the format in which the DHCP option data appears.

Although the DHCP library supports a few simple types (like integers, booleans, strings and IP addresses, as well as arrays of single types or fixed sequences of types), the fact that the data type carried by the 802.21 specific DHCP options is constituted by sub-options led to the definition of a new *"universe"*. An *"universe"* is the set of sub-options that belong to a data type. Table 5.4 represents the 802.21 specific DHCP options created.

Table 5.4: 802.21 specific DHCP options

Option	**Number**	**Universe**	**Data Type**
DHO_OPTION_IPv4_Address_MoS	139	MoS-ipv4-address	MoS_IS, MoS_CS, MoS_ES
DHO_OPTION_IPv4_FQDN_MoS	140	MoS-ipv4-fqdn	MoS_IS, MoS_CS, MoS_ES
DHO_OPTION_IPv6_Address_MoS	54	MoS-ipv6-address	MoS_IS, MoS_CS, MoS_ES
DHO_OPTION_IPv6_FQDN_MoS	55	MoS-ipv6-fqdn	MoS_IS, MoS_CS, MoS_ES

5.3.3 DNS-User

The DNS-User was implemented from scratch under a Linux environment and its only purpose is to discover PoS entities through the procedures defined in the RFC5679 [34]. Its class diagram, which is presented in the Figure 5.10, has the following components:

- ***DNS User:*** this component is responsible for the configuration of the domain name, as well as the registration with the MIHF and the management of the MIH signalling.
- ***DNS Control:*** is responsible for interacting with the DNS server, i.e., this component queries the DNS server for PoS, gathers the results and returns them to the *DNS User* component.
- ***Log:*** this component is defined as a singleton and it provides logging capabilities to the other components.

By default, the configuration of a domain name is necessary. Although the domain name can be extracted from *"resolv.conf"* file, the decision to make it configurable gives a greater control on which domain the discovery will take place. The DNS-User has two modes of discovery operations: standalone operation or complementary operation. In the standalone operation, the DNS-User discovers all needed information about the PoS in the configured domain name. In the complementary operation, the DNS-User operates as a second discovery mechanism, used to discover the remaining PoS information, i.e., the MIHF provides the domain name on which the DNS-User must discover PoS entities. The discovery process is triggered by the reception of a *MIH_Capability_Discover.indication* message and, based on the information presented, the DNS-User decides on which operation mode will execute the discovery of new PoS. The discovered entities are grouped in one single message and are forwarded to the MIHF through a *MIH_Capability_Discover.response* message.

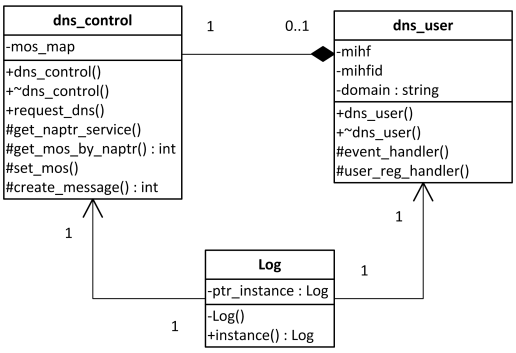

Figure 5.10: DNS-User class diagram

5.4 Summary

This chapter described the implementation of the local and remote discovery mechanisms, developed over an open-source 802.21 implementation. The proposed architecture is also depicted, as well as the implementation of the discovery entities.

In the next chapter the implementations of the discovery mechanisms, depicted in this work, are evaluated based on their performance, duration and information exchanged. The ODTONE remote communications are also evaluated to analyze their impact in the discovery mechanisms.

Chapter 6

Evaluation

This section presents the outcome of the discovery mechanisms implementation, providing a testbed description and the results obtained for the evaluation of the performance of ODTONE in signalling exchange, and the different implemented discovery mechanisms. ODTONE was evaluated regarding the configurations used for remote communications, i.e., based on the transport protocol and the MIH acknowledge service used. In what concerns the discovery mechanisms, the local discovery mechanism was evaluated based on its performance, while the remote discovery mechanisms were compared against each other.

6.1 ODTONE Performance Results

6.1.1 Scenario

ODTONE performance tests were deployed over two different machines of a physical testbed, the AMazING[1], simulating a MN and a PoA/PoS (Figure 6.1), which communicate with each other via Wi-Fi. Each one is composed by VIA Eden 1GHz processors with 1GB RAM, a 802.11abgn Atheros 9K and 802.11abg Atheros 5K radio interfaces and a Gigabit wired interface. Each node runs the Linux OS (Debian distribution) with kernel version 2.6.39-2-686-pae.

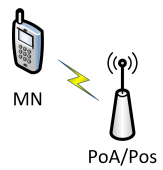

Figure 6.1: ODTONE performance testbed

This test aims to study the effect of different transport protocols and the MIH acknowledgement service in the remote communications. It consists in sending a generic MIH Request Message and to wait for the reception of the response. An environment with various entities sending messages at the same time was considered, so the time between the request and the response was measured for different incoming rates of request messages at the

[1]The AMazING (Advanced Mobile wIreless Network playGround) is a free access wireless tested, located at IT Aveiro rooftop. - http://amazing.atnog.av.it.pt/

MIHF. The results obtained are presented by average values with a 95% T-Student confidence interval.

6.1.2 Acknowledge Service

Figure 6.2 presents the results related to the combination of different transport protocols with the MIH acknowledge service.

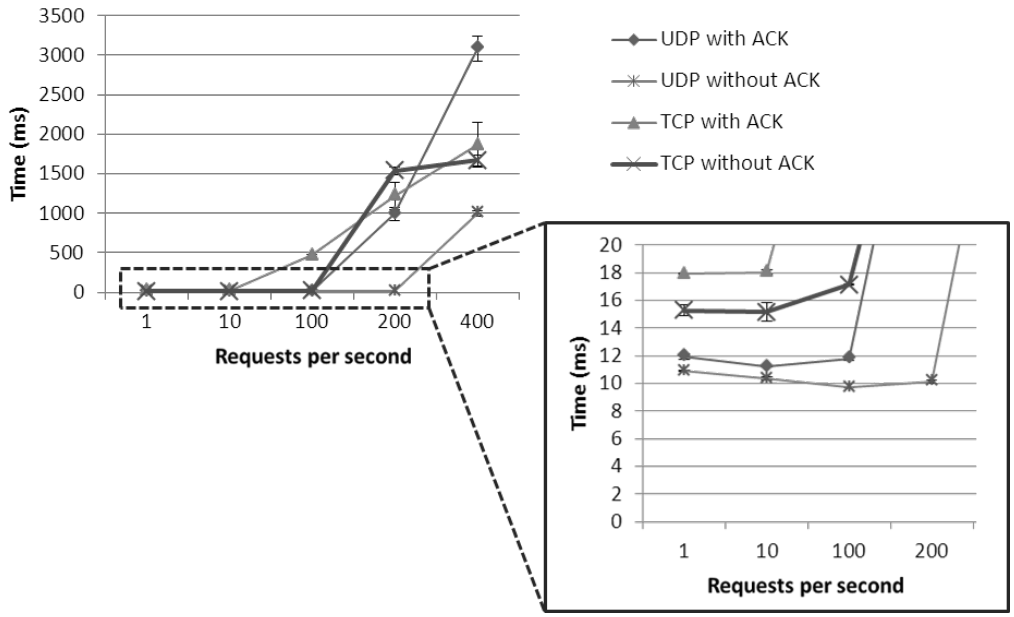

Figure 6.2: Acknowledge service time results

Analyzing the results obtained, the necessary time to complete a request/response transaction (for a single request) using UDP is about 10,9±0,035ms, which is increased by about 1ms when the MIH acknowledge service is enabled. When TCP is used instead of UDP, the results were a little higher, taking 17,9±0,057ms and 15,3±0,405ms for completing a transaction with and without the MIH acknowledge service enable respectively. The increase of time required to complete a transaction using TCP (compared with UDP) were expected due to the reliable data-transfer service of the TCP protocol. The slight increase of the transmission time using the MIH acknowledge service (either for UDP and TCP) occurs because of the increased overhead caused by the additional exchanged messages (MIH acknowledge messages) between the peer entities.

The transaction times remain constant up to values of 100 received request messages per second for UDP with the MIH acknowledge service and for TCP. For rates above these, an increase of the transaction time occurs, verifying that the longer the duration of the burst is, the greater is the time required to complete a transaction. This behaviour occurs early for the TCP with the acknowledge service (around 10 requests per second) and later for UDP (around 200 requests per second). When the average transaction time starts to increase, transaction loss is verified. This occurs due to message loss in the local communication between the MIHF and the Link SAP/MIH-User, which are not retransmitted.

The delay verified in the transaction completion happens when the incoming message rate is higher than the capacity of the MIHF to process them, leading to the queueing of the

messages in the receive socket buffer. Since the buffer has limited size (inherent to the Boost libraries), it results in packet loss if the buffer fills up before the messages are processed. By increasing the buffer size, although the average transaction time starts to increase at the same moment, the message loss occurs latter.

In terms of performance, RFC5677 [30] argues that it is expected that the MIH commands messages arrive at a rate of one in hundreds of milliseconds in order to capture quick changes in the environment and/or process handover commands. The obtained results show that, for all cases, the MIHF deployed is able to support the requirements, either for the MNs and the network entities. Therefore, the buffer can be configured properly to support the requirements of each entity.

In conclusion, the UDP protocol requires less time to complete a single request/response transaction than the TCP protocol, since the UDP protocol is a lightweight protocol that does not provide reliable data-transfer services. Also, the acknowledges sent by the MIH reliability mechanisms cause overhead so, its use is only useful for the UDP, which does not provide reliable services. Its use in the TCP is useless, since duplicated behaviour is added, which the TCP services already provide.

6.1.3 MIH Capability Discover Processing Performance

To further evaluate the impact of using MIH mechanisms in a discovery process, the MIHF performance when handling simultaneous capability requests from multiple nodes was measured. We have separately analyzed the time required to process the request and response messages, as well as to process the proposal for local discovery. As shown in Figure 6.3, a remote request takes longer to process (2.11±0,004ms) than the remote response (1.89±0,003ms), and the processing of a local request is the least expensive in terms of time (1.73±0,005ms). This behaviour occurs due to the fact that to process a remote request the MIHF creates an instance of the MIH state machine at the reception of the message and, since the response message is immediately available, the MIHF manages the state machine in order to send the response message. In addition to the response message, the MIHF also sends an indication message to the MIH-Users. The response message processing only accesses the state machine once and the processing of local requests bypasses the states machines. The creation and management of the MIH state machines, as well as the operations to send a message (which involves socket operations), are time-consuming processes, which increase the overhead in the MIHF, leading to delays in the total processing time of the messages.

The average processing time (for all cases) is constant up to values of 400 messages per second, above which a high increase of the average processing time as well as a loss of messages is verified. This decrease of performance occurs when the incoming message rate is higher than the time required to process a single message, leading to the queueing of the message in the receive socket buffer. Since the buffer has a limited size, it results in message loss if the buffer fills up before the messages being processed. Increasing the size of the receive socket buffer, for the same tests, the MIHF was able to queue more messages and, therefore, message loss was not verified.

The MIH Capability Discover mechanism is shared by all remote discovery mechanisms presented in this work (with results presented in the following section), and therefore, the previous results influence on their performance.

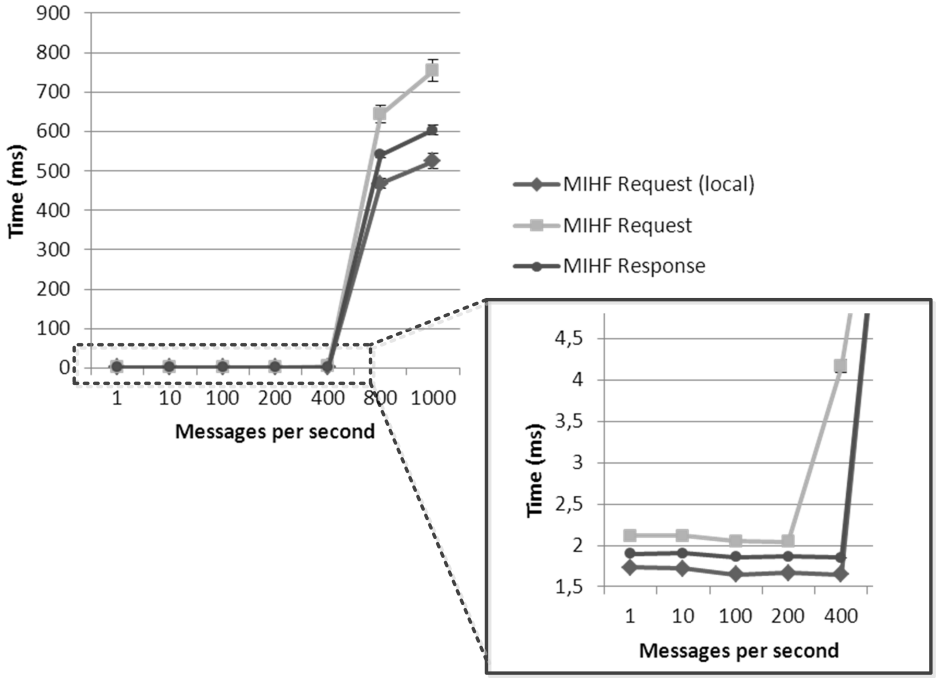

Figure 6.3: MIH Capability Discover MIHF processing time

6.2 Discovery Mechanisms Results

6.2.1 Scenario

To evaluate the performance of the different discovery mechanisms, their implementations were deployed on a physical testbed, the AMazING. It is constituted by 24 fixed nodes, each one composed by VIA Eden 1GHz processors with 1GB RAM, a 802.11abgn Atheros 9K and 802.11abg Atheros 5K radio interfaces and a Gigabit wired interface. Each node runs the Linux OS (Debian distribution) with kernel version 2.6.39-2-686-pae.

Figure 6.4 represents the L3 and L2 scenarios used to deploy and evaluate our implementation.

The L3 testbed is composed by a MN, a PoS and DNS/DHCP servers. The MN is connected to the network via a Wi-Fi AP and the link between the AP, the PoS and the DNS/DHCP servers is composed by Gigabit Ethernet. In this scenario, the MN tries to discover the available PoS making use of each one of the L3 discovery mechanisms described in this dissertation. The L2 testbed consists of a MN and 10 equidistant APs. Here, the MN is not associated with any AP and communications are made via L2 Management Frames, using the L2 discovery mechanisms. For the local discovery scenario, a single machine, simulating a MN, is used to evaluate the discovery of the local entities. All local communications were made using the UDP protocol without acknowledgements, while for the remote communications between peer MIHF the UDP protocol with acknowledgement service was used.

In all scenarios, the MN and the PoS have the default components associated to IEEE 802.21: a MIHF, a MIH-User and two Link SAPs (it is assumed that the PoS and MN have two network interfaces). In addition, the MN still has a DHCP-User and a DNS-User, which will be responsible for interacting with the DHCP and DNS servers respectively. The

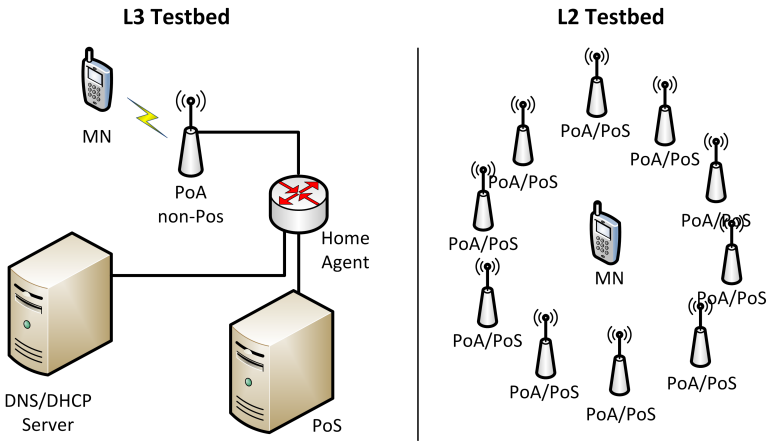

Figure 6.4: L3 and L2 testbeds

configuration of each entity for each scenario was done with the help of configuration files and scripts (see Annex B).

Each discovery mechanism was tested for several hundred times, presenting average results with a 95% T-Student confidence interval. They are evaluated according to the amount of time required, as well as the information size involved. These represent two important factors to be considered in time-restricted procedures (such as handovers) and the load imposed in network procedures, respectively. These results contain both MIH-User/Link SAP to MIHF local interaction and the remote interaction between MIHF peers.

6.2.2 Local Discovery Mechanisms Evaluation

The proposed local discovery mechanisms (described in the section 4.1) were subject to an intensive analysis for measuring their impact in the device bootstrap. In the test scenario, the Link SAP registration mechanisms required the exchange of 110 bytes of information and it took about 2,2±0,002ms to complete the whole process. By comparison, the local MIH-User registration with the MIHF required the exchange of 21 bytes of information and 0,5±0,001ms. The higher amount of information and the longer time of execution occur for Link SAP registration because the MIHF requests the capabilities of the Link SAPs, whereas this is not needed in the MIH-User registration.

The request of the Link SAPs capabilities when they register enables the MIHF to maintain its local capabilities constantly updated. Therefore, the MIHF is able to provide its capabilities right away when a capability request from a peer entity (or from upper layers) is received, instead of requesting it locally every time. The performance of both situations in a remote MIH Capability Discover was measure in terms of time required, exchanged messages and amount of information (Table 6.1). Comparing the results, it was concluded that having the capabilities stored at the MIHF allows saving at least 3,5ms and 135 bytes of information per capability discover transaction, due to the nonexistent interaction with the Link SAPs.

An additional scenario simulating requests done by a MIH-User to a Link SAP was tested. Contrary to the default behaviour, the proposed local scheme allows the MIHF to maintain up-to-date awareness of the Link SAP activity (i.e., on or off due to energy conservation procedures while scanning for PoAs), thus avoiding the propagation of unnecessary messages. For this simulation, a network interface alternates between active and inactive states, based

Table 6.1: Comparison between the storage of the local MIHF capabilities in a remote MIH Capability Discover transaction

	With storage	**Without storage**
Time (ms)	8,26±0,20	11,78±0,16
Exchanged messages	5	9
Bytes	262	387

on a Poisson distribution with a mean duration of 4s. Each simulation run takes 60s, with the interface remaining active for an average of 70% of the simulation time.

Figure 6.5 depicts the simulation results obtained by varying the number of request messages received per second. Wasted bytes represent the amount of information sent towards the interface while inactive, while Updated bytes represent our informational events updating the Link SAP state to the MIHF.

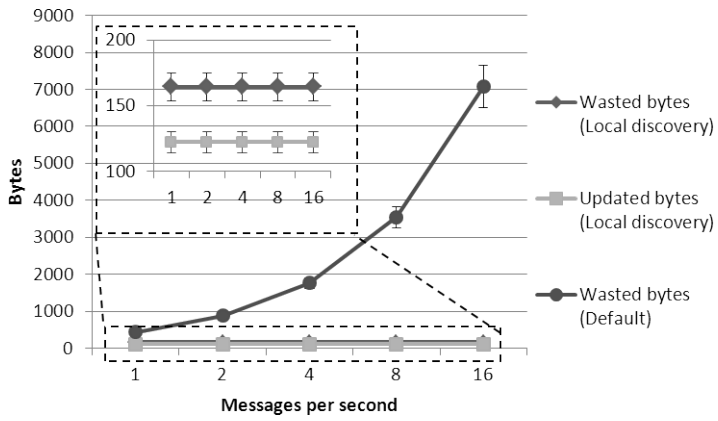

Figure 6.5: Amount of missed Link SAP requests comparison

Comparing the obtained results, the default procedure creates a large amount of information in wasted messages sent to the inactive Link SAP, in contrast to our local discovery procedure. For its part, our local discovery procedure wastes the bytes of one single request message to detect that the Link SAP becomes inactive (164 bytes in average), requiring, however, to update its activation with the MIHF (122 bytes in average).

6.2.3 Remote Discovery Mechanisms Comparison

The experimental results for the total duration (both L2 and L3) as well as the amount of information exchanged in each remote discovery mechanism is depicted in the Table 6.2. It contains the total time required and amount of exchanged information of each discovery mechanism, presenting the percentage of involvement that the MIH processes have in the discovery. Figure 6.6 shows the portion related to the MIH interaction, against other involved protocols, such as DHCP and DNS.

For the MIH L2 Beacon, also L2 Listening to Enhanced Media Dependent Beacon (described in the section 4.2.1), the results concern the time between the scan trigger and receiving responses at the MN, i.e., the measured time corresponds to the scanning time. The

Table 6.2: Remote Discovery Mechanisms Comparison

	Duration		Information Exchanged	
Method	**Total Time (ms)**	**MIH %**	**Total Size (bytes)**	**MIH %**
MIH L2 Beacon	1970±90	100,0	75	100,0
L2 Capability Discover	5,8±0,2	100,0	308	100,0
Broadcast	8,8±0,1	100,0	308	100,0
DNS	38,9±1,4	27,3	1161	37,7
DHCP	23,4±0,2	47,6	1002	40,1
DHCP+DNS	44,8±1,3	23,1	1852	28,6
DHCP Bootstrap	304,9±17,4	5,9	1641	26,9
DHCP Bootstrap+DNS	308,9±15,2	5,4	2491	22,8

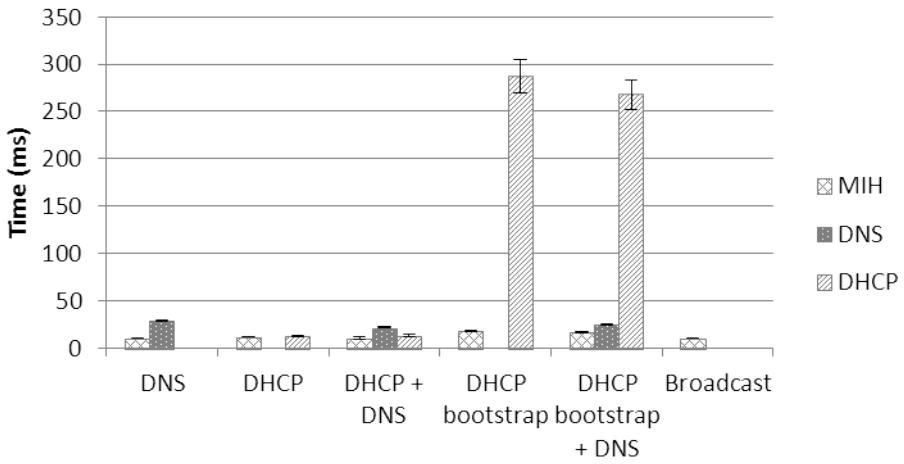

Figure 6.6: L3 discovery mechanisms information type comparison

average scanning time is about 1,97±0,09s, making it the mechanism that demands more time. However, this time is completely dependent on the OS, the hardware and technology scanning process. Regarding the information exchanged, this mechanism requires the transport of the least amount of MIH information. This is due to the nonexistence of a MIH request sent by the MN (i.e., the PoA/PoS is always sending its capabilities embedded in the Beacon Frames with no prior MIH Capability Discover Request received). The L2 Capability Discover Exchange (described in the section 4.2.1) allowed the fastest discovery time, about 5,8±0.2ms. Although it exchanges the same amount of MIH information than the L3 Broadcast Discovery procedure, the L2 mechanism is faster because it avoids IP routing and the processing of an additional layer of the protocol stack. Both L2 discovery mechanisms only exchange MIH information for the PoS discovery (thus 100% of MIH time and information), and (along with the L3 Broadcast method) are the methods which use the least amount of data in overall, since no additional discovery procedures (such as DNS and/or DHCP) are required.

For L3 mechanisms, the L3 Broadcast (described in the section 4.2.2.1) is the fastest (8,8±0,1ms) and it is the least information demanding L3 mechanism, since it only exchanges MIH information. However, it can only be used in scenarios where the PoS and the MN are in the same network domain. When the PoS is in a different network than the MN, the DHCP and/or DNS mechanisms (described in the section 4.2.2.2 and 4.2.2.4 respectively) need to be included in the discovery process. These additional mechanisms increase the amount of time and exchanged information of the discovery process. If compared with the MIH protocol exchanges, DNS exchanges take approximately 3 times longer and requires 2,5 times more information, while DHCP exchanges take about the same amount of time and information than the MIH exchanges. The DHCP results are greatly affected when the PoS discovery is done at the bootstrap (described in section 4.2.2.3), reaching values 17 and 3 times higher for the mechanism duration and information exchanged respectively, when compared to the MIH protocol. Results show that DHCP bootstrap with DNS discovery (described in the section 4.2.2.5) is not only the most time consuming discovery mechanism but also the one that requires the largest amount of information to be transported. This is mostly due to the DHCP phase, which has to deal not only with PoS discovery but also with the IP address configuration of the MN, because of its occurrence at the bootstrap phase. When using DHCP and DNS, the footprint of the MIH signalling is always lower than the corresponding L3 discovery mechanism, particularly in the cases where both are used, since they compose the majority of the information exchanged between entities in those situations (close to 78,3% and 52,4%, respectively, and 76,9% when both are used, when not in bootstrap).

The discovery mechanisms that only use MIH procedures proved to be the fastest. However, they are limited in terms of locations, i.e., for the L2 scenarios these are only capable of discovering the PoS at the L2 coverage range, while in the L3 scenarios they can only discover PoS that belong to the same multicast group. Thus, the integration with other mechanisms, such as DHCP and DNS, enables a larger scope in terms of locations with the higher cost of discovery time.

These results were made over a controlled scenario and hence an increased delay on the discovery times can happen in real scenarios. It depends on several network factors, such as location of the DHCP/DNS servers and PoS, routing, overhead or even QoS configurations. Local communications are not included in the previous assumption. Therefore, Table 6.3 makes a contrast between the amount of information exchanged locally and remotely, for each entity involved in the discovery process.

All messages exchanged for DHCP and DNS interactions are remote. However, in what concerns the MIH interactions, the messages exchanged can be local or remote. Analyzing the results, the remote MIH exchanges represent 147 bytes of information in all scenarios. This value corresponds to the MIH Capability Discover messages exchanged by the MN and the PoS to discover each other capabilities. Thus, the DHCP/DNS communications with the respective server, as well as the remote MIH Capability Discover messages exchanged between the MN and the PoS, are the ones that most affect the increase of the delay in the discovery process.

6.3 Summary

The implementation is evaluated according the results of the discovery mechanisms implementation, as well as, the performance results of the ODTONE. The ODTONE

Table 6.3: Comparison between remote and local information exchange

		MIH Messages (bytes)	DNS Messages (bytes)	DHCP Messages (bytes)	Total (bytes)
DNS	**Local**	291	0	0	291
	Remote	147	723	0	870
DHCP	**Local**	255	0	0	255
	Remote	147	0	600	747
DHCP + DNS	**Local**	382	0	0	382
	Remote	147	723	600	1470
DHCP Boot.	**Local**	294	0	0	294
	Remote	147	0	1200	1347
DHCP Boot. + DNS	**Local**	421	0	0	421
	Remote	147	723	1200	2070
Broadcast	**Local**	161	0	0	161
	Remote	147	0	0	147

implementation was evaluated on its capability to process request/response messages, making a more detailed analysis on the capability of the MIHF to process the MIH Capability Discover transactions. The discovery mechanisms were evaluated based on their duration and amount of transmitted information, comparing each other results.

In the next chapter, the key point of the developed work is concluded, highlighting the main contributions points and points out further work.

Chapter 7

Conclusion

In this chapter the developed work is summarized, mentioning the problems experienced during its development and the main contributions for the research area. Finally, open problems for future research are identified, focusing on IEEE 802.21, which was the denominator key of the whole work.

7.1 Issues

With the development of discovery mechanisms, several issues have been identified in what concerns technology and OS independence. The first issue is related with the implementation of L2 mechanisms. These are dependent of the technology, since each technology has its own messages and mechanisms to communicate. In addition, network device management is also dependant on the OS, once it requires certain functions and/or capabilities from the host OS.

The development of L3 discovery mechanisms, more specifically the DHCP mechanisms, also proved to be dependent on the OS. The dependency on the OS is due to the fact that the DHCP mechanism needs to configure the network information in the interface. This requirement is similar to the requirements of the L2 mechanisms.

In the end, this work aims to overcome these problems by offering an abstract way to communicate with the MIHF, either by providing a common interface for all mechanisms or by defining the L3 discovery entities as MIH-Users. Still, each L2 or L3 specific mechanism must be implemented.

7.2 Main contributions

In this work three main contributions can be identified: the evaluation of the existing MIH discovery mechanisms, the proposal of a novel mechanism to discover local MIH entities and the integration of the IEEE 802.21 with the PMIPv6.

The MIH discovery mechanisms proved to be a solution that overcomes the restrictions imposed by static configurations. The dynamic environment supported by these mechanisms enables the deployment of several different scenarios requiring minimum configurations. In this way, the discovery of the entity that controls mobility in the network (e.g., when the MN has no prior knowledge about it or even the recovery from the single point of failure of the current PoS) is one of the scenarios that become possible with these mechanisms. They can be done at L2, by using enhanced media specific messages with MIH information, and at

L3, by using DHCP or DNS services. These mechanisms can be used in an isolated way or complementary through the combination of several discovery mechanisms.

Actually, several devices (such as laptops) support hot-plug of new NICs, allowing their dynamic attachment and detachment from the system while they are running. The proposed local discovery mechanism allows the MIHF to be aware of these changes, enabling the attachment and detachment of new Link SAPs without having to reboot. The mechanisms defined for the local discovery have also proved to be a solution that improves the performance of the MIH Capability Discover, since it locally stores the updated capabilities with no need to determine the existence of the Link SAPs each time a request is received.

Furthermore, this work have contributed to this research area with an article published on the IEEE Communications Magazine (September 2011), named "Using an open-source IEEE 802.21 implementation for network-based localized mobility management" [4]. The proposed integration scenario (presented in Annex C) was able to demonstrate the flexibility of IEEE 802.21 to support not only MN-controlled but also network-controlled scenarios. Its coupling with the PMIPv6 mobility protocol provides the necessary features to fully operate in heterogeneous environments, having an abstract way of interacting with multiple technologies. In addition, by having link-layer triggers regarding MN attachment, the IEEE 802.21 solves some of the out of scope mechanisms presented in the PMIPv6 standard. Results show that a network-controlled mobility scenario is able to support the IEEE 802.21 signalling footprint, when used to support and enhance the procedures of the PMIPv6.

A paper entitled "Evaluation of Discovery Mechanisms for Media Independent Handover Services" was also submitted for the 2nd IEEE Workshop on Convergence among Heterogeneous Wireless Systems in Future Internet. In what concerns the ODTONE project, several improvements and extensions have been developed, as well as, bug fixing and support to questions on the mailing lists and to Advanced Telecommunications and Networks Group (ATNoG) members that, under other projects, needed the ODTONE. One of these projects is the MultiMEDia transport for mobIlE Video AppLications (MEDIEVAL)[1].

7.3 Future work

With the continuous growth of multi-technology operator solutions, IEEE 802.21 will play a major role in near future communications through the provision of MIH mechanisms for information retrieval and link layer control.

7.3.1 Discovery Mechanisms implementation

DNS Improvements

We believe that the DNS-User performance can be improved by storing the queries results during their lifetime, allowing to share the results with multiple requests. Also, the DNS-User can be configured to use the additional records presented on the DNS response messages in order to avoid additional requests to the DNS Server.

[1]MEDIEVAL - http://www.ict-medieval.eu/

Remote MIHF Capabilities Storage

Although the current implementation stores the peer MIHF capabilities, there is no mechanism that makes use of them. Therefore, if the capabilities of the peer MIHF are already known, the MIH Capability Discover remote exchange can be bypassed. However, it may be necessary mechanisms to ensure that the capabilities are up-to-date.

7.3.2 IEEE 802.21

Security

Without security, MIH entities and MIH services are vulnerable to several security attacks, such as tampering, replay attacks, eavesdropping and even MoS identity spoofing. Currently, the security of the MIH protocol relies on the underlying transport protocols security mechanisms, which does not provide authentication procedures and therefore, authentication must be take into consideration to protect the MIH communications. However, the increased overhead due to the security signalling can significantly increase the latency of the handover, which may, in some cases, make service continuity impossible. Thus, network access authentication and key establishment mechanisms must be optimized in order to minimize the latency in the handover caused by the security messages exchange [48].

Broadcast Handover

Currently, no standard specifies mechanisms to support handovers between downlink-only technologies. With the widespread use of the downlink-only technologies (such as Terrestrial Digital Multimedia Broadcasting (T-DMB)), optimized handover between these technologies have been subject of research, berthing an amendment to IEEE 802.21, IEEE 802.21b [49].

Multicast

Multicast is considered an ideal mechanism to transmit multimedia contents, based on the group communication and IP multicasting. Mobile multicast applications may require fast network switching with QoS guarantee and, therefore, mobile networks should be design to support multicast service. To achieve this purpose, its integration with MIH services [50] is being studied and evaluated.

Hierarchical MIIS

Currently, the definition of a single MIIS server in the network is considered on the available works. However, this architectural assumption has several problems [51]:

- It represents a single point of failure.
- High MIIS discovery times depend on the location of the MN.
- Too much information stored in a single entity.

The research on this field aims for a solution that enables the storing of network information in an hierarchical way, by splitting the existing information among different MIIS.

Bibliography

[1] E. Gustafsson and A. Jonsson. Always best connected. *Wireless Communications, IEEE*, 10(1):49 – 55, February 2003.

[2] C. Perkins, D. Johnson, and J. Arkko. Mobility Support in IPv6. RFC 6275 (Proposed Standard), July 2011.

[3] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil. Proxy Mobile IPv6. RFC 5213 (Proposed Standard), August 2008.

[4] D. Corujo, C. Guimaraes, B. Santos, and R.L. Aguiar. Using an open-source IEEE 802.21 implementation for network-based localized mobility management. *Communications Magazine, IEEE*, 49(9):114 – 123, September 2011.

[5] J.F. Kurose and K.W. Ross. *Computer networking: a top-down approach*. Addison-Wesley, 2010.

[6] C. Perkins. IP Mobility Support for IPv4, Revised. RFC 5944 (Proposed Standard), November 2010.

[7] IEEE standard for local and metropolitan area networks - part 21: Media independent handover. *IEEE Std 802.21-2008*, pages c1 – 301, 2009.

[8] IEEE standard for information technology- telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std 802.11-1997*, pages i – 445, 1997.

[9] I.F. Akyildiz, Jiang Xie, and S. Mohanty. A survey of mobility management in next-generation all-ip-based wireless systems. *Wireless Communications, IEEE*, 11(4):16 – 28, August 2004.

[10] A.T. Campbell, J. Gomez, Sanghyo Kim, Chieh-Yih Wan, Z.R. Turanyi, and A.G. Valko. Comparison of ip micromobility protocols. *Wireless Communications, IEEE*, 9(1):72 – 82, February 2002.

[11] E. Fogelstroem, A. Jonsson, and C. Perkins. Mobile IPv4 Regional Registration. RFC 4857 (Experimental), June 2007.

[12] H. Haverinen and J. Malinen. Mobile IP Regional Paging. Internet-Draft draft-haverinen-mobileip-reg-paging-00, Internet Engineering Task Force, June 2000. Work in progress.

[13] C.E. Perkins and D.B. Johnson. Route optimization for mobile ip. *Cluster Computing*, 1:161 – 176, April 1998.

[14] R. Koodli. Mobile IPv6 Fast Handovers. RFC 5568 (Proposed Standard), July 2009.

[15] C. Perkins, D. Johnson, and J. Arkko. Mobility Support in IPv6. RFC 6275 (Proposed Standard), July 2011.

[16] Tran Cong Hung, Le Phuc, Tran Thi To Uyen, Hae Won Jung, and Yoohwa Kang. Improving handover performance in mobile ipv6. In *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on*, volume 3, pages 1828 – 1831, February 2008.

[17] Dong cheol Shin and Sung gi Min. Fast handover solution using multi-tunnel in hmipv6 (fm-hmipv6). In *Sensor Technologies and Applications, 2008. SENSORCOMM '08. Second International Conference on*, pages 833 – 838, August 2008.

[18] F.Z. Yousaf and C. Wietfeld. Optimizing the performance of fmipv6 by proactive proxy bindings. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 – 5, September 2009.

[19] T. Melia, J. Korhonen, R.L. Aguiar, S. Sreemanthula, and V. Gupta. Network initiated handovers problem statement. Internet-Draft draft-melia-mipshop-niho-ps-00, Internet Engineering Task Force, June 2006. Work in progress.

[20] H. Jang, A. Yegin, K. Chowdhury, and J. Choi. Network initiated handovers problem statement. Internet-Draft draft-ietf-mip6-hiopt-17.txt, Internet Engineering Task Force, May 2008. Work in progress.

[21] K. Chowdhury, M. Khalil, and H. Akhtar. Home Subnet Prefix or the Home Agent discovery for Mobile IPv6. Internet-Draft draft-chowdhury-mipv6-home-prefix-00.txt, Internet Engineering Task Force, April 2004. Work in progress.

[22] R. Droms. Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6. RFC 3736 (Proposed Standard), April 2004.

[23] R. Droms, J. Bound, B. Volz, T. Lemon, C. Perkins, and M. Carney. Dynamic Host Configuration Protocol for IPv6 (DHCPv6). RFC 3315 (Proposed Standard), July 2003. Updated by RFCs 4361, 5494, 6221.

[24] K. Omae, I. Okajima, and N. Umeda. Mobility anchor point discovery protocol for hierarchical mobile ipv6. In *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, volume 4, pages 2365 – 2370 Vol.4, March 2004.

[25] M. Liebsch, A. Singh, H. Chaskar, D. Funato, and E. Shim. Candidate Access Router Discovery (CARD). RFC 4066 (Experimental), July 2005.

[26] T. Melia and Y:. El Mghazli. Access Network Service Discovery Function discovery. Internet-Draft draft-melia-radext-andsf-discovery-extension-00, Internet Engineering Task Force, January 2009. Experimental.

[27] W. Song, Jong-Moon Chung, Daeyoung Lee, Chaegwon Lim, Sungho Choi, and Taesun Yeoum. Improvements to seamless vertical handover between mobile wimax and 3gpp utran through the evolved packet core. *Communications Magazine, IEEE*, 47(4):66 – 73, April 2009.

[28] S. Frei, W. Fuhrmann, A. Rinkel, and B.V. Ghita. Improvements to inter system handover in the epc environment. In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, pages 1 – 5, February 2011.

[29] K. Doppler, C.B. Ribeiro, and J. Kneckt. On efficient discovery of next generation local area networks. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, pages 269 – 274, March 2011.

[30] T. Melia, G. Bajko, S. Das, N. Golmie, and JC. Zuniga. IEEE 802.21 Mobility Services Framework Design (MSFD). RFC 5677 (Proposed Standard), December 2009.

[31] D. Corujo. IEEE 802.21 in heterogeneous handover environments. Master's thesis, Universidade de Aveiro, Aveiro, Portugal, 2007.

[32] B. Aboba, M. Beadles, J. Arkko, and P. Eronen. The Network Access Identifier. RFC 4282 (Proposed Standard), December 2005.

[33] R. Elz and R. Bush. Clarifications to the DNS Specification. RFC 2181 (Proposed Standard), July 1997. Updated by RFCs 4035, 2535, 4343, 4033, 4034, 5452.

[34] G. Bajko. Locating IEEE 802.21 Mobility Services Using DNS. RFC 5679 (Proposed Standard), December 2009.

[35] G. Bajko and S. Das. Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Options for IEEE 802.21 Mobility Services (MoS) Discovery. RFC 5678 (Proposed Standard), December 2009.

[36] P. Machan, S. Serwin, and J. Wozniak. Performance of mobility support mechanisms in a heterogeneous umts and IEEE 802.11 network offered under the IEEE 802.21 standard. In *Information Technology, 2008. IT 2008. 1st International Conference on*, pages 1 – 4, May 2008.

[37] T. Melia, D. Corujo, A. de la Oliva, A. Vidal, R.L. Aguiar, and I. Soto. Impact of heterogeneous network controlled handovers on multi-mode mobile device design. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 3884 – 3889, March 2007.

[38] L. Eastwood, S. Migaldi, Qiaobing Xie, and V. Gupta. Mobility using IEEE 802.21 in a heterogeneous IEEE 802.16/802.11-based, imt-advanced (4G) network. *Wireless Communications, IEEE*, 15(2):26 – 34, April 2008.

[39] T. Melia, A. de la Oliva, A. Vidal, I. Soto, D. Corujo, and R.L. Aguiar. Toward ip converged heterogeneous mobility: A network controlled approach. *Computer Networks*, 51(17):4849 – 4866, 2007.

[40] Xinyi Wu and Gang Nie. Design and simulation of an enhanced handover scheme in heterogeneous mobile ipv6 networks. In *Information Processing, 2009. APCIP 2009. Asia-Pacific Conference on*, volume 2, pages 448 – 451, July 2009.

[41] Yoon Young An, Byung Ho Yae, Kang Won Lee, You Ze Cho, and Woo Young Jung. Reduction of handover latency using mih services in mipv6. In *Advanced Information Networking and Applications, 2006. AINA 2006. 20th International Conference on*, volume 2, pages 229 – 234, April 2006.

[42] Yufei Feng, Yumei Wang, and Lin Zhang. A novel pmipv6-based mobility management scheme using IEEE 802.21 for converged network. In *Network Infrastructure and Digital Content, 2010 2nd IEEE International Conference on*, pages 465 – 469, September 2010.

[43] T. Cardoso, P. Neves, M. Ricardo, and S. Sargento. Media independent handover management in heterogeneous access networks - an empirical evaluation. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 – 5, May 2011.

[44] S. Mansor and Tat-Chee Wan. Mobility management in heterogeneous wireless access network with IEEE 802.21 services. In *Computer and Network Technology (ICCNT), 2010 Second International Conference on*, pages 110 – 114, April 2010.

[45] T. Melia. Mobility Services Transport: Problem Statement. RFC 5164 (Informational), March 2008.

[46] libnl - netlink protocol library suite @ONLINE, October 2011.

[47] nl80211 - linux wireless @ONLINE, October 2011.

[48] IEEE draft standard for local and metropolitan area networks: Media independent handover services - amendment for security extensions to media independent handover services and protocol. *IEEE P802.21a/D04*, pages 1 – 85, August 2011.

[49] IEEE draft standard for media independent handover services - amendment: Handovers with downlink only technologies. *IEEE P802.21b/D04*, pages 1 – 85, August 2011.

[50] In-Seop Jang, Won-Tae Kim, and Yong-Jin Park. An efficient mobile multicast mechanism based on media independent handover. In *Communications (MICC), 2009 IEEE 9th Malaysia International Conference on*, pages 501 – 505, December 2009.

[51] F. Buiati, L.J.G. Villalba, D. Corujo, J. Soares, S. Sargento, and R.L. Aguiar. Hierarchical neighbor discovery scheme for handover optimization. *Communications Letters, IEEE*, 14(11):1020 – 1022, November 2010.

[52] M.Q. Khan and S.H. Andresen. An intelligent scan mechanism for 802.11 networks by using media independent information server (miis). In *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*, pages 221 – 225, March 2011.

[53] J.M. Arraez, M. Esseghir, and L. Merghem-Boulahia. An implementation of media independent information services for the network simulator ns-2. In *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, pages 492 – 496, January 2011.

Annex A

ODTONE - An Open-Source IEEE 802.21 Implementation

ODTONE stands for Open Dot Twenty ONE and it is an open source implementation of the Media Independent Handover framework from the IEEE 802.21 Media Independent Handover Services standard, using C++ APIs. Beside the MIHF, it also provides a set of APIs to enable ODTONE users to implement their own MIH Users and Link SAPs and to interface with ODTONE MIHF.

A.1 ODTONE's MIHF Architecture

As presented in Figure A.1, which depicts the ODTONE's MIHF architecture, the ODTONE's MIHF implements the three core MIH services, each containing a set of logical components. The *MIES* module allows the MIHF to verify if the received event messages are formatted according to the standard, which MIH-Users have subscribed the events and, if applicable, to forward the message to the subscribed MIH-Users. These represent, respectively, the roles of the *Event validator*, *Event subscriber* and *Event publisher* modules. The *MICS* module also provides a way to validate the received message and to forward them to their destination, operations that are in charge of the *Command validator* and *Command publisher* modules. The definition of the IS is out of the scope of the standard. In ODTONE, and following the approach taken by several works [52] [53], the IS acts as a MIH-User and therefore, the *MIIS* module is responsible to forward the messages to the IS registered with the MIHF or, if is that the case, forward the response message to the requestor.

The previously mentioned modules allow the MIHF to provide the basic features of the MIH protocol. Additionally, the ODTONE's MIHF architecture has other components which allow not only to implement the remaining features of the MIH protocol but also to add robustness to the MIHF:

- **Service Manager (SM):** this module is responsible for the management of the MIH Capability Discover messages, which provide information about the services supported by an MIH peer.

- **Communication handler:** collects the messages received from different SAPs or peer MIHFs entities and forwards them to the *Service Access Controller* module.

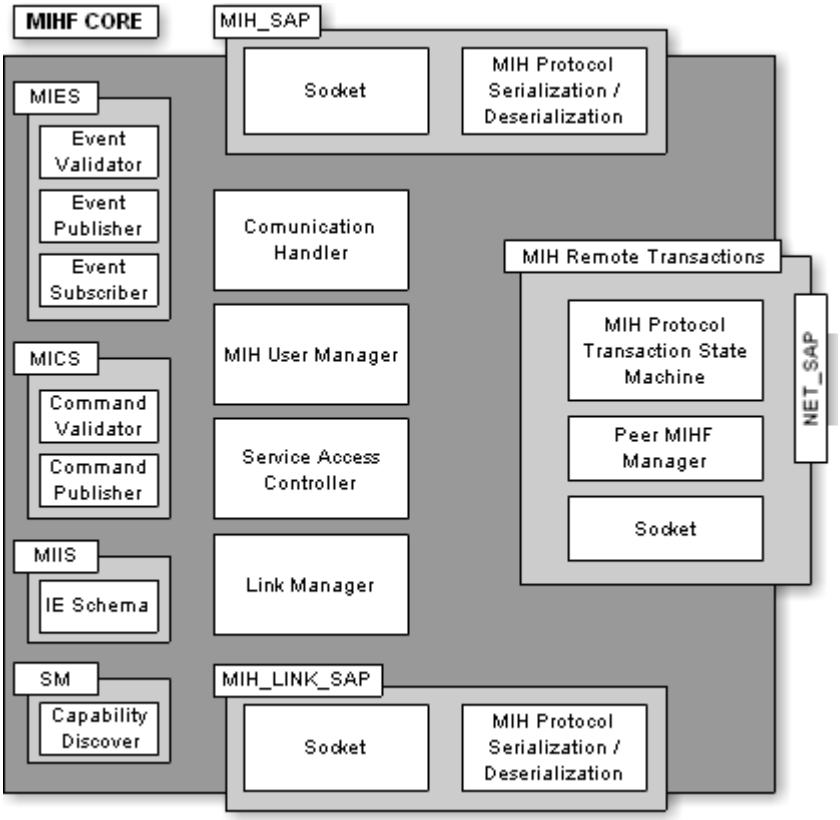

Figure A.1: ODTONE's MIHF architecture

- **Service access controller:** responsible for analyzing the header of the MIH messages and to forward them to the respective MIH service. The decision is based on the MIH message identifier and the registered callbacks.
- **Link manager:** stores the information about the available Link SAPs.
- **MIH-User manager:** provides the MIHF with information about the registered MIH-Users.
- **Peer MIHF manager:** provides the MIHF with the transport information and the capabilities of the peer MIHFs.
- **Transaction state machine controller:** keeps the state of each remote transaction with peer MIHF entities. It is responsible, if applicable, for sending the acknowledge messages and to detect the reception of duplicates messages.

A.2 Achieving OS Independence

One of the main objectives of ODTONE project is to implement a MIHF that is capable of being deployed in multiple operating systems, i.e., that is independent of the OS.

The OS independence is achieved by using the Boost libraries. These allow the network work-level operations and the definition of datatypes that are system-independent. For

network work-level operations, ODTONE uses Boost.Asio library, which is based on the proactor pattern, while for the definition of datatypes, Boost.Variant and Boost.Optional are used, facilitating the definition of the MIH datatypes and handling of the messages. Boost.System, Boost.Thread, Boost.Build and Boost.Quickbook allow the management of system error codes, threads, building and documentation to be done in a portable way.

Also, ODTONE, instead of traditionally implementing the SAPs as a software API, reuses the MIH protocol, used for remote communications between peer MIHFs, to provide local communications between the SAPs and the MIHF. Thus, the MIH-Users and Link SAPs are decoupled from the MIHF, allowing them to be coupled with high-level and link-level software, respectively, as long as they respect the MIH protocol. To support this feature, the MIH primitives need to be mapped into MIH messages. In order to facilitate this procedure, ODTONE provides a library featuring all datatypes and primitives, which can be used by the SAPs and the MIHFs to interact with each other. Nevertheless, the decoupled architecture of the MIHF and its SAPs allows the SAPs to be implemented in other languages, as long as they conform to the MIH protocol. This facilitates the plug-in of entities interfacing the MIHF, in different languages and OSs, but also the integration of new access technologies beyond those specified in the standard.

A.3 Extensions

The work developed in this dissertation enabled the extension of ODTONE to support the discovery mechanisms depicted, providing the MIHF the ability to discover the available PoS. The implementation of these mechanisms takes into account the objectives and assumptions of the ODTONE project.

To support the proposed local discovery, the MIHF was extended to recognize two new messages and act in accordance with the proposed mechanism. Regarding the remote discovery mechanisms, a new sub-module was implemented in the SM, which is responsible for managing the signalling related to the discover procedures. Also, the MIHF was extended to provide a unified interface for the L2 discover procedures, allowing the attachment of different implementations. This work provided a 802.11 driver for Linux. In what concerns the L3 discovery entities, two new discovery entities were implemented: the DNS-User and the DHCP-User. These entities act as MIH-Users, which are responsible to communicate with its server in order to obtain the PoS information. Its communications with the MIHF is made using sockets, due to the architecture adopted by the ODTONE, which reuses the MIH protocol in the local communications. In addition, the implementation of these mechanisms led to several changes in the library provided by the ODTONE, as well as, in several MIHF modules (see chapter 5 for more details).

Annex B

Testbed configuration

B.1 L3 Testbed

The L3 testbed (B.1) is composed by a MN, a PoS and DNS/DHCP servers. The MN is connected to the network via a Wi-Fi AP and the link between the AP, the PoS and the DNS/DHCP servers is composed by Gigabit Ethernet.

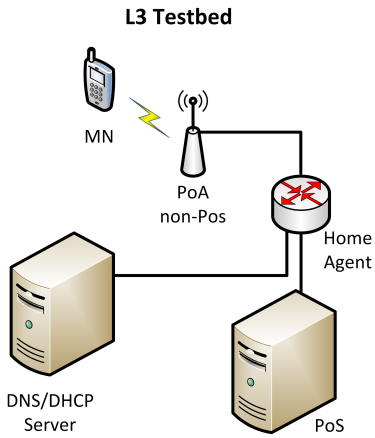

Figure B.1: L3 testbed

B.1.1 DHCP server

DHCP Server	option domain-name ”odtone.test”;
	option domain-name-servers ns1.odtone.test, ns2.odtone.test;
	default-lease-time 600;
	max-lease-time 7200;
	log-facility local7;
	subnet 192.168.1.0 netmask 255.255.255.0 {
	range 192.168.1.10 192.168.1.20;
	option option-ipv4-address-mos_ip.is 10.10.01.01, 10.2.1.2;
	option option-ipv4-address-mos_fqdn.is odtone.test;
	}

B.1.2 DNS server

DNS Server	$ORIGIN odtone.test. $TTL 1W @ 1D IN SOA localhost. root.localhost. (2006101001 ; serial 3H ; refresh 15M ; retry 1W ; expiry 1D) ; minimum 1D IN NS ns ns 1D IN A 127.0.0.1 _MIHIS._tcp 1D SRV 0 0 4551 mos _MIHES._udp 1D SRV 0 0 4551 mos _MIHCS._udp 1D SRV 0 0 4551 mos mos.odtone.test. 1D IN A 10.110.1.5 odtone.test. 1D IN A 127.0.0.1 odtone.test. 1D IN NAPTR 50 50 "s" "MIHIS+M2T" "" _MIHIS._tcp odtone.test. 1D IN NAPTR 50 50 "s" "MIHES+M2U" "" _MIHES._udp odtone.test. 1D IN NAPTR 50 50 "s" "MIHES+M2U" "" _MIHCS._udp

B.1.3 PoS

MIHF	[mihf] id = mos.odtone.test local_port = 1025 remote_port = 4551 80211_listener = ath1
Link SAP 1	[link] id=link1 port = 1235 tec = 802_11 link_addr_list = 00:11:22:33:44:55 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025
Link SAP 2	[link] id=link2 port = 1236 tec = 802_11 link_addr_list = 00:11:22:33:44:11 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025

Note: The MIHF and Link SAPs configurations are relative to ODTONE.

B.1.4 PoA

hostapd	interface=ath1 driver=nl80211 logger_syslog=-1 logger_syslog_level=2 logger_stdout=-1 logger_stdout_level=2 dump_file=/tmp/hostapd.dump ctrl_interface=/var/run/hostapd ctrl_interface_group=0 ssid=odtone1 hw_mode=g channel=1 beacon_int=100 dtim_period=2

B.1.5 MN

Broadcast

MIHF	[mihf] id = mihf1 local_port = 1025 remote_port = 4551

DHCP

MIHF	[mihf] id = mihf1 discover_order = DHCP local_port = 1025 remote_port = 4551

DNS

MIHF	[mihf] id = mihf1 discover_order = DNS local_port = 1025 remote_port = 4551

DHCP + DNS

MIHF	[mihf] id = mihf1 discover_order = DHCP, DNS local_port = 1025 remote_port = 4551

Note: The MIHF configuration is relative to ODTONE.

B.2 L2 Testbed

The L2 testbed (B.2) consists of a MN and 10 equidistant APs. The MN is not associated with any AP and communications are made via L2 Management Frames, using the L2 discovery mechanisms.

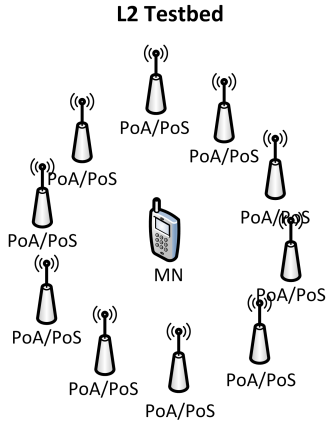

Figure B.2: L2 testbed

MN

MIHF	[mihf] id = mihf1 local_port = 1025 remote_port = 4551 80211_listener = ath1
Link SAP 1	[link] id=link1 port = 1235 tec = 802_11 link_addr_list = 44:11:22:33:44:55 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025
Link SAP 2	[link] id=link2 port = 1236 tec = 802_11 link_addr_list = 00:11:22:77:44:11 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025

Note: The MIHF and Link SAPs configurations are relative to ODTONE.

PoA/PoS

hostapd	interface=ath1 driver=nl80211 logger_syslog=-1 logger_syslog_level=2 logger_stdout=-1 logger_stdout_level=2 dump_file=/tmp/hostapd.dump ctrl_interface=/var/run/hostapd ctrl_interface_group=0 ssid=odtone1 hw_mode=g channel=1 beacon_int=100 dtim_period=2
MIHF	[mihf] id = mos1.odtone.test local_port = 1025 remote_port = 4551 80211_listener = ath1
Link SAP 1	[link] id=link1 port = 1235 tec = 802_11 link_addr_list = 00:11:22:33:44:55 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025
Link SAP 2	[link] id=link2 port = 1236 tec = 802_11 link_addr_list = 00:11:22:33:44:11 event_list = link_detected, link_up, link_down, link_parameters_report, link_going_down, link_handover_imminent, link_handover_complete [mihf] ip=127.0.0.1 local_port=1025

Note: This is an example how to configure a PoA/PoS. Thus, the remaining PoA/PoS differ on the *SSID* and *channel* in the *hostapd* configuration file, and in the MIHF ID in the MIHF configuration file. The MIHF and Link SAPs configurations are relative to ODTONE.

Annex C

Integrating PMIPv6 with IEEE 802.21

During the work developed for this dissertation, a scenario of a mobility protocol integrated with the IEEE 802.21, beyond those described in the standard, is proposed and evaluated[1], to prove the extensibility of this framework to support new mobility protocols. It introduces a new entity in the network, which the MN must previously discover in order to be able to use the services provided by it.

C.1 Integration Scenario

The PMIPv6 standard does not specify mechanisms to detect the attachment and detachment of MNs to the MAGs. In addition, the IEEE 802.21 standard, per itself, does not perform handover actions and therefore a mobility management protocol needs to be associated. This presents an opportunity to exploit the mechanisms provided by each one of the standards and to integrate them in order to achieve an optimized network-based localized mobility management.

Figure C.1 proposes the signalling for network controlled seamless handover achieved by integrating the PMIPv6 and the IEEE 802.21 protocols [4]. This integration is enhanced with media independent commands for resource availability check, preparation and release, as well as event indication concerning link status and commands to trigger PMIPv6 specific actions.

This proposal introduces a new entity, named Mobility Decision Engine (MDE), that acts as a PoS and which is responsible for making network controlled handover decisions. The discovery mechanisms discussed in this work have a key role in the discovery of the network controlling node, i.e., the MDE. Therefore, once the MN attaches to the network, it can use the discovery mechanisms in order to the MN and the MDE discover each other.

The MDE receives indications from the MN and network PoAs and controls the PMIPv6 handover process, supported by IEEE 802.21 signalling. The information concerning the network topology is stored in the Information Server (IS) that can be queried by the MDE to gather information about PoAs that are nearby a given MN. Also, the PMIPv6 client residing at the MAGs acts as an MIH-User that is connected to the MIHF.

Initially the MN is connected to the MAG1 and within the coverage area of the MAG2. Both MAGs belong to the same operator and to the same PMIPv6 domain. Several triggers

[1] A part of this chapter is based on the work published in [4]

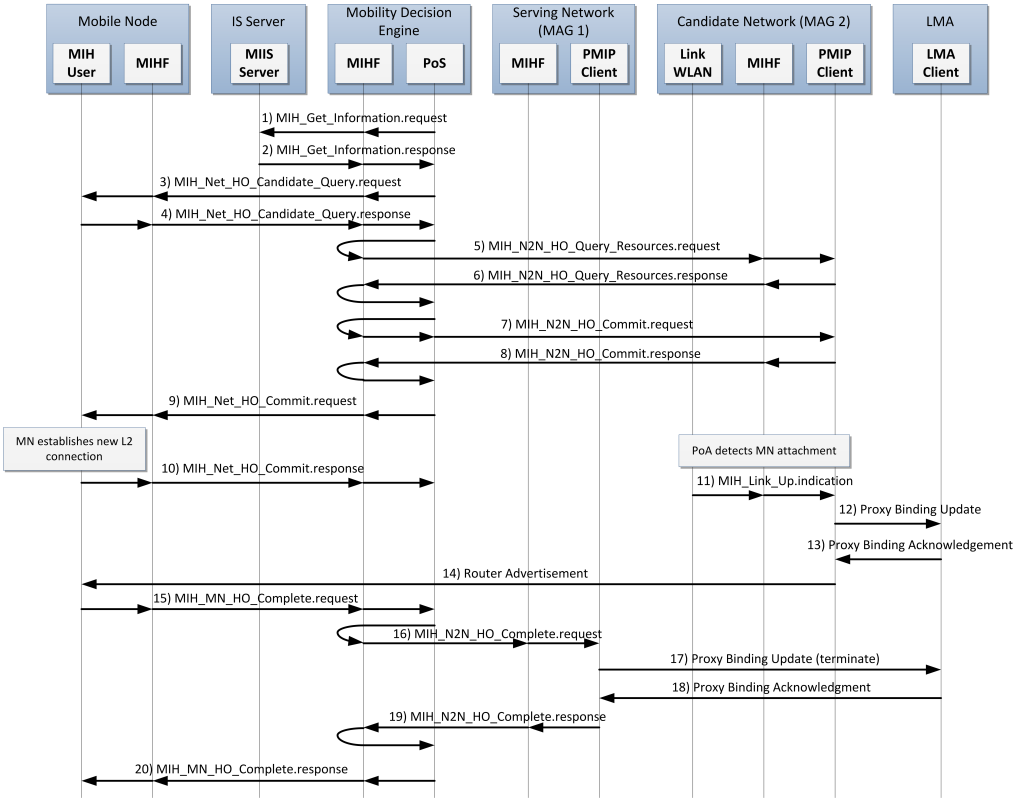

Figure C.1: PMIPv6 and IEEE 802.21 integration signalling

(that are not defined in this scenario) enable the MDE to detect that the MAG1 is not the best access network in the range of the MN, prompting the MDE to query the IS for information about nearby networks (step 1). Based on the information retrieved by the IS, the MDE initiates the handover by sending the candidate networks to the MN, that satisfy its requirements (step 3). The MN indicates its preferred access network (step 4) and, based on that, the MDE queries the resources available in the selected candidate network (steps 5 and 6) and, if they are available, the MDE commits them (steps 7 and 8). After the resources have been allocated, the MDE instructs the MN to handover to the selected access network (step 9). Upon the L2 attachment to the MAG2, the MN notifies the MDE about its result (step 10). Parallel to this, the MAG2 detects the L2 attachment of the MN (step 11) and notifies it to the PMIP client, which is then responsible to register the location of MN and to update the tunnel towards the new MN location by triggering a PBU message towards the LMA (step 12). Upon the reception of the Router Advertisement message (step 14), the MN has the necessary information to configure its IP address on the interface. At this moment, the tunnel to the LMA is updated towards the MAG2 and the handover procedure has been concluded. Thus, the MN informs the MDE that it has finished the handover procedures (step 15), which is then responsible to release the committed resources (step 16) and previous bindings on the old connection (steps 17 and 18). Finally, the MN is acknowledged about handover completion (step 20).

In this scenario, during the handover procedure a seamless experience is provided, since the MN connects to the candidate network while keeping the old connection. Only when the

new connection is established, the MN releases of the old connection.

C.2 Evaluation

C.2.1 Scenario

To evaluate the PMIPv6 and IEEE 802.21 integration scenario (Figure C.2), six nodes (each one representing one of the entities) were deployed on the AMazING testbed. Each node is composed by VIA Eden 1GHz processors with 1GB RAM, a 802.11abgn Atheros 9K and 802.11abg Atheros 5K radio interfaces and a Gigabit wired interface, running Ubuntu Linux 11.04 environment. The communications related to the MN are made via Wi-Fi, while the rest of the communications is performed via wired links. In what concerns the IEEE 802.21, the ODTONE implementation was used, while for the PMIPv6 mechanisms, the OPMIP[2] implementation was chosen.

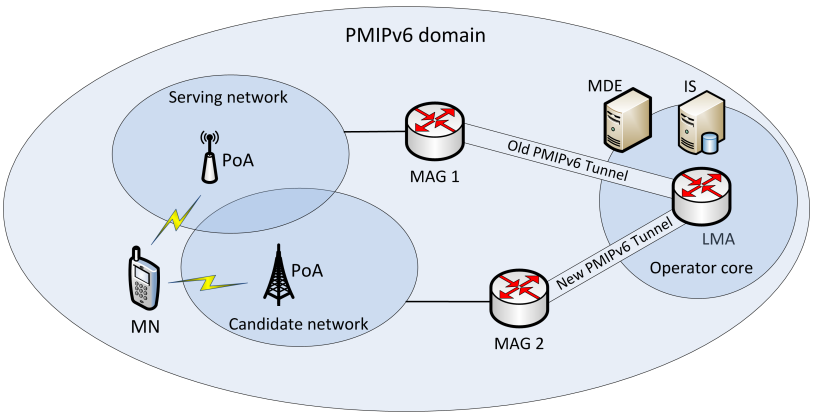

Figure C.2: PMIPv6 and IEEE 802.21 integration testbed

The obtained results aim to evaluate the ODTONE's footprint in the proposed signalling framework. In this way, network transmission and LMA times are not accounted for, since network transmission will depend on network deployment conditions and LMA is only involved in PMIPv6 signalling. The results are presented by average values with a 95% T-Student confidence interval.

C.2.2 PMIPv6 with IEEE 802.21 integration results

The evaluation of the integration of PMIPv6 with 802.21 focuses on the footprint of each entity in the mobility process, regarding the amount of information exchanged (Table C.1) and the processing time of each interaction (Table C.2) belonging to the interactions involving ODTONE's components. The aim is to evaluate the impact caused by the 802.21 signalling in this handover scheme.

While analyzing the results, it was noticed that the MDE is the most active element in the whole process, getting values of 10,50ms and 1984 bytes of exchanged information, which represents 44% and 48% of the total signalling time and transmitted information

[2]OPMIP is an Open-source implementation of the Proxy Mobile IP Mobility Management Protocol. This implementation is fully based on the IETF's RFC 5213. - http://atnog.av.it.pt/projects/opmip

Table C.1: Exchange information in the ODTONE and PMIPv6 integration scenario

Exchanged information (bytes)			
Total			4444
Per entity	**Mobile node**		556
	Mobility Decision Engine		1984
	Mobile Access Gateway 1		187
	Mobile Access Gateway 2		651
	Media Independent Information Server		1066
Per message	**IEEE 802.21 Commands**	**Information Service**	1871
		Candidate Query	415
		Resource Query	497
		Resource Commit	391
		Handover Initiation	381
		Handover Complete	687
	IEEE 802.21 Events		202

respectively. Most of this exchanged information belongs to the IS query (about 42%), due to the RDF schema size in the query reply. This step can be optimized by filtering mechanisms implemented at both the MIH-User of the MDE and the MIIS, reducing the response size, or basing only the handover candidate selecting in scanning procedures of the MN. When comparing these results with the ones obtained by the MN, the MN takes less time (21%) and exchanges less information (13%). The communications that involve the MN are the only ones that are made via Wi-Fi and, therefore, the low value of exchanged information performed by the MN is a good result concerning the limitations of wireless transmissions compared with the wired ones. These results were expected, since this scenario demonstrated a network-based mobility control where the MN assists the candidate query and handover completion processes. This indicates the advantage of network controlled mobility decisions to reduce the MN participation, which is relevant in what concerns battery consumption efficiency.

Comparing the results of each MAG, the MAG1 (which is the actual serving MAG) is the entity that has less involvement in the handover process (3% and 4% of the total signalling time and transmitted information respectively), while the signalling that involves the MAG2 represents 15% of the total signalling time and 19% of the total transmitted information. The higher values of the MAG2 highlight its involvement in the resources querying and committing processes, as well as detecting the MN attachment for triggering the PMIPv6 procedures.

In conclusion, these communications times represent an acceptable trade-off for a system able to be deployed in different mobility scenarios and independent of specific interfaces available in the MN.

Table C.2: Processing time in the ODTONE and PMIPv6 integration scenario

Processing Time (ms)			
Total			21,51 ±0,005
Per entity	**Mobile node**		4,64±0,002
	Mobility Decision Engine		10,50±0,004
	Mobile Access Gateway 1		0,77±0,003
	Mobile Access Gateway 2		4,12±0,002
	Media Independent Information Server		1,48±0,004
Per message	**IEEE 802.21 Commands**	**Information Service**	2,77±0,003
		Candidate Query	3,21±0,002
		Resource Query	3,06±0,002
		Resource Commit	2,84±0,002
		Handover Initiation	3,07±0,001
		Handover Complete	5,18±0,001
	IEEE 802.21 Events		1,37±0,002