UNIVERSITÁ DI PISA
Facoltá di Ingegneria

Corso di Laurea Specialistica in INGEGNERIA INFORMATICA
Curriculum NETWORKING E MULTIMEDIA

Network-Assisted Handover for Heterogeneous Wireless Networks using IEEE 802.21

Tesi di
Francesco Galeassi

Anno Accademico
2009 - 2010

Relatori:	Prof. Luciano Lenzini	....
	Prof. Enzo Mingozzi	....
	Dr. Claudio Cicconetti	....
Candidato:	Francesco Galeassi	....

Abstract

The IEEE 802.21 is a standard for enabling handover in heterogeneous wireless networks. Published in January 2009, it defines protocols and messages for mobile-to-node and node-to-node communication in a technology-neutral and flexible manner. The need arises because of the widespread diffusion of different technologies for wireless communications (e.g., WiFi, WiMAX, LTE) coexisting in the same geographical area. Even though the number of multi-radio multi-technology mobile devices is increasing significantly, there are no open solutions in the market to enable efficient inter-technology handover.

As is often the case with communication standards, the structure of the required components, the procedures, and the algorithms are left unspecified by the IEEE 802.21 standard so as to promote competion by differentiation of equipment capabilities and services. The contribution of this thesis is two-fold: i) a design and an implemenation of the Media Independent Information Service (MIIS) server; and, ii) a solution to enable network-assisted handover using the IEEE 802.21 standard, aimed at reducing the handover latency and the energy consumption of mobile devices due to scanning.

The MIIS server has been fully implemented in C++ under Linux. In order to perform testbed evaluations, all the required components have been implemented, as well, within an open source framework for IEEE 802.21 called ODTONE. Modifications to the latter have been performed for optimization and fine tuning, and for extending those functional modules needed but not fully implemented.

For a realistic evaluation, Linux-based embedded COTS devices have been used, equipped with multiple IEEE 802.11a and IEEE 802.11g wireless network interface cards. This has required additional development for kernel/user space binding and hardware control.

Testbed results are reported to show the effectiveness of the proposed solution, also proving the MIIS server scalability.

Sommario

Lo standard IEEE 802.21 permette l'handover trasparente tra reti wireless eterogenee. Pubblicato nel Gennaio 2009, definisce il protocollo ed i messaggi necessari alla comunicazione tra i dispositivi mobili e la rete in maniera flessibile ed indipendente dalla tecnologia usata. Questo bisogno nasce dalla notevole diffusione nella stessa area geografica di diverse tecnologie per la comunicazione wireless (come ad esempio, WiFi, WiMAX e LTE). Nonostante stia crescendo significativamente il numero di dispositivi mobili muniti di più interfaccie radio con tecnologie differenti, ad oggi non c'è nessuna soluzione disponibile sul mercato che permetta una transizione efficiente tra le diverse tecnologie

Come spesso succede negli standard di comunicazione, il documento IEEE 802.21 non definisce né la struttura dei vari componenti di rete, né le procedure e gli algoritmi necessari per un handover efficiente tra le diverse tecnologie, in modo tale da promuovere la competizione tra i prodotti e i servizi forniti da aziende concorrenti. Il contributo di questa tesi è duplice: viene proposta, i) l'architettura e l'implementazione del Media Independent Information Service (MIIS) server; e ii) una soluzione basata sullo standard IEEE 802.21 per abilitare l'handover assisitito dalla rete, in modo tale da ridurre il tempo necessario per effettuare l'handover e l'energia consumata dal nodo mobile durante la scansione.

Il MIIS server è stato completamente implementato sotto Linux in C++. Sono state inoltre implementate tutte le componenti necessarie per effettuare i test sperimentali, utilizzando il framework open source per lo standard IEEE 802.21 chiamato ODTONE. Sono state necessarie alcune modifiche al framework stesso, al fine di ottimizzare ed estendere i moduli funzionali necessari, che non erano stati completamente implementati.

Per una valutazione realistica della soluzione proposta, sono stati utilizzati dispositivi embedded basati su Linux equipaggiati con interfacce wireless IEEE 802.11a e IEEE 802.11g. Questo ha richiesto uno sviluppo ulteriore per permettere la comunicazione kernel/user space ed il controllo dell'hardware stesso.

Sono infine riportati i risultati sperimentali, per dimostrare l'efficacia della soluzione proposta e la scalabilità del MIIS server.

Ai miei genitori, a mia sorella e a Michela.
Ai miei amici, quelli veri.

Ed ai miei nonni.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In the recent past wireless broadband Internet connectivity has been massively deployed. Nowadays, there is a widespread diffusion of different technologies for wireless communications (e.g., Wireless Fidelity (WiFi), Long Term Evolution (LTE) and Worldwide Interoperability for Microwave Access (WiMAX)) coexisting in the same geographical area. Even though the number of multi-radio multi-technology mobile devices is increasing significantly, there are no open solutions in the market to enable efficient inter-technology handover. As a consequence, the wireless landscape will remain diverse for the foreseeable future, making heterogeneity an important factor to address. This scenario is creating a strong demand for smooth intra-technology (horizontal) and inter-technology (vertical) handover solutions so as to keep the mobile device connected to the most suitable access network at a given time and place.

In addition, handover procedures must allow service providers to implement handover policies based on a variety of operational and business requirements. In this environment, a key challenge is to ensure handover performance, especially when considering real-time services such as video or voice applications (e.g., Voice over IP (VoIP) and Internet Protocol Television (IPTV)). In fact, if one of the above applications is running when handover occurs, and the handover is not seamless, users will be dissatisfied. In all such cases, service continuity should be maintained to the extent possible during handover.

The definition of the words smooth and seamless is "*with little or no perceptible disruption of the user's applications*". This requirement is particulary stringent for real-time or streaming applications, but best effort services should also operate satisfactorily in presence of handovers. The major goal of the seamless handover is to hide either from the application or the user any differences between the normal service offered within a domain and during a migration. This notion of transparency is, however, very much dependent on the service being provided, since each application has a specific set of properties which need to be satisfied during all service time. As an example, when making a network transition during a cell phone call, handover procedures should be executed in such a way that any perceptible interruption to the conversation will be minimized.

Connecting laptops to broadband access at hotels and WiFi hotspots has become a standard part of everyday life, particularly for international business travellers. In addition, many travellers utilize smartphones and broadband connections for voice communications. Using VoIP services when making international calls, rather than the Public Switched Telephone Network (PSTN), can lead to noteworthy savings on roaming costs. This represents a significant opportunity for mobile operators.

Figure 1.1 shows an example scenario with heterogeneous overlapping wireless networks. Imagine that a multiaccess mobile device user makes a VoIP call as he moves in this area. First, he is at home where Internet access is provided by a WiFi Access Point (AP) and he starts the call. When he leaves the home and travels around the city, he will not need to break down this active call if the network and the Mobile Node (MN) support seamless handover. Outside Internet connection is guaranteed by WiMAX network. When the user arrives at the office, the MN automatically switches from WiMAX to the available WiFi network in a smooth way. In fact, free WiFi network is a better solution if, for instance, the user pays based on traffic volume and he does not have a flat-rate plan. The advantages of WiMAX is the bigger coverage area and its Quality Of Service (QoS) guarantees.

Given the diversity of network applications running on mobile devices, knowledgeable network resource planning and operation is needed, in turn calling for a framework that allows users and their applications to state their network access preferences. This framework should also allow operators to design terminal access patterns aiming at maximizing resource usage and increasing user satisfaction. Currently, this process can only be done manually: users need to be aware of available access networks and to choose the one they want to connect with, on the basis of very rudimentary information such as signal quality. If MNs might collect timely and consistent information about the state of all available networks in range and control their network connectivity, then a whole range of possibilities would

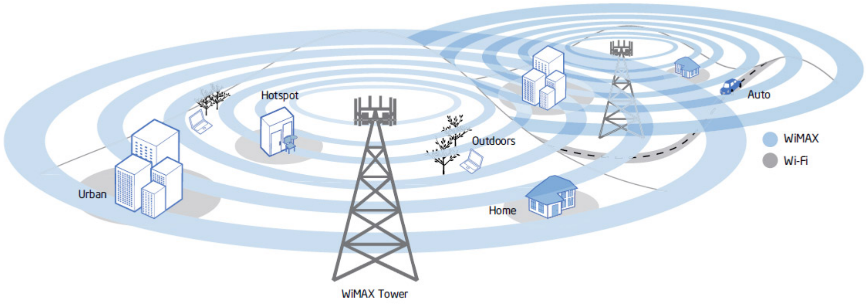

Figure 1.1: Example scenario with heterogeneous overlapping wireless networks

become available. Operators who can switch a user's session from one access technology to another could better manage their networks and better accomodate service requirements of their users.

For example, when the quality of an application running on the network decreases, the application may be transferred to another less congested network with a higher throughput. Operators might also leverage this ability to manage multiple interface so as to balance traffic loads across available networks more appropriately, improving system performance and capacity. In order to optimize the use of available network resources, MNs need to be able to collect information about numerous heterogeneous networks in a generic and standardized way, irrespective of the underlying network access technology. The collected information, both dynamic and static, might then be used by handover decision-making processes. These may be enhanced versions of Mobile IP (MIP), proprietary solutions or other proposals, such as [1].

Researchers in the area have proposed several cross-layer frameworks for enhancing the efficiency of handover decision makers [1] [2], but none of them has been formally standardized or is widely accepted so far. One way to ensure handover interoperability across multiple access technologies is to create mupltiple, media-specific extensions. This approach, in fact, is a scalable and efficient method of addressing inter-technology handovers. With a common media-independent framework in place, each access technology requires only a single extension to ensure interoperability with all other technologies. The complexity of this approch grows in line with the number of elements (O(N)) and scales more efficiently than a media-specific approach (O(N^2)). So what is needed is a standard framework that can attract ample support from major vendors and operators, and that can be deployed incrementally. To address this need, the Institute of Electrical and Electronics Engineers (IEEE) 802.21 standard [3] for Media Independent Handover (MIH) services has been published in January 2009.

1.2 IEEE 802.21 Working Group

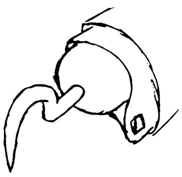

Figure 1.2: IEEE 802.21 logo

The IEEE 802.21 Working Group (WG) started work in March 2004. The group produced a first draft of the standard including the protocol definition in May 2005. Figure 1.3 illustrates the progress toward the IEEE 802.21-2008 standard. The latest draft version of the standard was accepted as a new standard by the

IEEE-SA Standards Board in November 2008. The standard was published in January 2009. IEEE 802.21 defines the mechanisms to enable and optimize handover between heterogeneous IEEE 802 networks, including those where handover is not otherwise defined, and facilitate handover between IEEE 802 networks and cellular networks, as well as Fourth Generation (4G) wireless networks [4]. This goal is achieved by defining new entities and services that must be implemented into the mobile and the network devices and an extensible communication protocol.

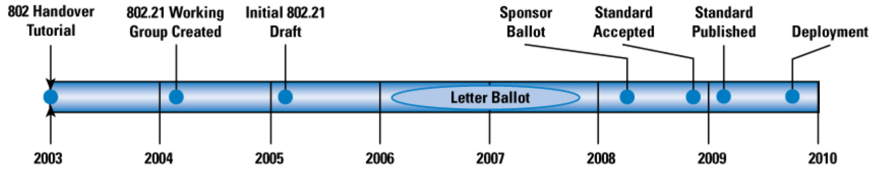

Figure 1.3: IEEE 802.21 history and timeline

The IEEE 802.21 standard defines a framework to facilitate handover between heterogeneous access networks by:

- Reporting events from the lower layers.
- Defining commands to control the functions of the lower layers.
- Exchanging information about available link types, link identifiers and link qualities of nearby network links.

Moreover, it helps with handover initiation and handover preparation, while handover execution is out of the scope of the standard and must be handled by higher layer mobility management protocols (e.g., MIP and Session Initiation Protocol (SIP)). Both network-initiated and mobile-initiated handovers are supported, enabling co-operative handover decision making.

For example, the handover process can be initiated by mesaurement reports (e.g., signal quality and transmission error rates) and triggers supplied by the link layers on the MN. The network, having a complete view of the neighbor acces networks, can then supply the MN with the most suitable access network, so as to complete the handover process.

Note that the scope of IEEE 802.21-2008 is restricted to access technology-independent handovers. Intra-technology handovers, handover policies, security mechanisms, media-specific link layer enhancements to support IEEE 802.21-2008 and higher layer (Layer 3 (L3) and above) enhancements which are required to support this standard are outside the scope of IEEE 802.21-2008. To fulfil some topics left unspecified in the standard, new Task Groups (TGs) are been developing:

- **IEEE 802.21 TG “a”**, which is developing security extensions to MIH services to reduce the latency during authentication and the key establishment for handovers between heterogeneous access networks that support IEEE 802.21. Another objective is to provide data integrity, replay protection, confidentiality and data origin authentication to IEEE 802.21 MIH protocol exchanges and enable authorization for MIH services.

- **IEEE 802.21 TG “b”**, which is developing handovers for downlink only technologies to defines mechanisms that enable the optimization of handovers between IEEE 802.21 supported technologies and Downlink-Only (DO) technologies such as Digital Video Broadcasting (DVB), Terrestrial Digital Multimedia Broadcasting (T-DMB) and Media Forward Link Only (MediaFLO).
- **IEEE 802.21 TG “c”**, which is developing optimized single radio handover solutions to enable optimized single radio handovers between heterogeneous IEEE 802 wireless technologies and extend these mechanisms for single radio handovers between IEEE 802 wireless technologies and cellular technologies.

The IEEE 802.21 standard is described in detail in Chapter 2. For a gentle introduction to the standard, refered to [5], [6], [7], [8].

1.2.1 Competing technologies

WiOptiMo[1] is an application layer solution based on Java which enables any mobile device to pick the best connection to the Internet among those available in any time and place, to preserve it in case of temporary absence of the signal (e.g., a tunnel), to seamlessly switch to a better one as soon as it becomes available (e.g., a free WiFi hotspot) and to scale back to a less performing one when the connection in use is no more available [9]. During a connection hole (e.g., in a tunnel) and/or during a switch, the Internet session which is active gets frozen and restarts seamlessly as soon as the connection is again available. Compared with IEEE 802.21 standard, WiOptiMo operates at Layer 7 (L7) of the Open Systems Interconnection (OSI) protocol stack, creating an application layer tunnel between user processes, so it does not provide a standard media-independent framework for enabling seamless handover. Moreover, this solution is proprietary so it may hardly attract ample support from major vendors and operators.

Unlicensed Mobile Access (UMA)/Generic Access Network (GAN)[2] is the Third Generation Partnership Project (3GPP) global standard for subscriber access to mobile circuit, packet and IP Multimedia Subsystem (IMS)-based services over any Internet Protocol (IP)-based access network, including the Internet. This standard provides the mechanism for mobile operators to extend their services over any broadband IP access network. With GAN, all mobile services (voice, data, Internet) are routed over WiFi and the broadband access network. As a result, mobile operators can leverage WiFi as a seamless extension of the mobile network, delivering high-quality, low-cost mobile services to subscribers at home, in the office and in

---

[1] http://www.wioptimo.com/wioptimo/.

[2] http://www.smart-wi-fi.com/index.php.

hotspots. Thus mobile operators can provide smartphone subscribers a GAN client in the form of a software application. Compared with IEEE 802.21 standard, this solution is oriented only to mobile phones interacting with WiFi hotspots, without considering any other wireless technologies as WiMAX.

1.3 Proposed solution

As is often the case with communication standards, the structure of the required components, the procedures and the algorithms are left unspecified by the IEEE 802.21 standard so as to promote competion by differentiation of equipment capabilities and services.

The contribution of this thesis is two-fold: i) a design and an implemenation of the Media Independent Information Service (MIIS) server; and, ii) a solution to enable network-assisted handover using the IEEE 802.21 standard, where the mobile devices are instructed by the network itself on the best possible serving access points, thus removing the need for scanning. This way both the handover latency and the energy consumption of the mobile device are reduced, and benefits are especially significant if multiple radios coexist in the same device, since only one of them needs to be powered on at any time.

A proof-of-concept experimental evaluation, with two overlapping non-interoperable IEEE 802.11a and IEEE 802.11g networks, has been carried out to show the feasibility and effectiveness of the proposed solution.

1.4 Related Work

The motivation for a network-assisted handover is provided by the previous work in the literature, though not involving the IEEE 802.21 standard. For instance, the benefits in terms of handover delay and fair service satisfaction have been demonstrated through testbed experiments in [10], at least for the case of homogeneous IEEE 802.11 networks. On the other hand, the impact of scanning is investigated in [11], where the authors propose an analytical model to predict the successful scan probability in a heterogeneous IEEE 802.11/IEEE 802.16 network, also taking into account some practical service charge functions.

With regard to the specific area of IEEE 802.21, in [12] the authors present a simulation of handover between Third Generation (3G) and Wireless Local Area Networks (WLANs), observing a trade-off between service continuity and WLAN utilization. Analysis and simulation have also been employed in [13] [14] to verify that the overall signaling can be significantly reduced by employing the IEEE 802.21 capabilities in vehicular networks.

With regard to practical implementations, the early work in [15] describes a prototype implementation of a Mobility Manager for handling not only handover, but also power management and adaptation. However, the Mobile IPv6 for Linux (MIPL) was used instead of a pure IEEE 802.21 because the latter was still in an early draft stage. In [16] is proposed a framework for the implementation of the IEEE 802.21 MIH standard and its performances are evaluated through experiments in integrated IEEE 802.11/802.16e networks. From the experimental results in real testbeds, the authors demonstrate that the MIH services can be used to reduce packet losses, the APs discovery time and the energy consumption of mobile nodes during a vertical handover.

More recently, practical considerations for implementing IEEE 802.21 in Linux have been discussed in [17], which also includes a blueprint of the functional blocks required. This implementation focuses on the support of several access technologies including IEEE 802.11, IEEE 802.16 and IEEE 802.3 using off-the-shelf Linux capabilities.

The approach of [18] is complementary and aims at providing a generic MIH services framework, encompassing the standard's communication model, transport and state machines. An illustrative scenario for proactive pre-authentication using Commercial, Off-The-Shelf (COTS) devices is also presented to demonstrate how network selection and handover preparation can be leveraged by such an implementation. A pre-authentication method combined with IEEE 802.21 is also presented in [19]. In [20] is illustrated the design and implementation of a MIH middleware on a Linux platform so as to support a high quality VoIP system.

Finally, in [21] the authors present a vertical handover framework built around the concept of IEEE 802.21 and complemented by modules inspired by IEEE 1900.4 [22] for context information collection and efficient handover decision making and control.

1.5 Structure of the thesis

The rest of the thesis is structured as follows. A detailed description of the IEEE 802.21 standard is provided in Chapter 2. In Chapter 3 is analyzed the problem statement, some handover scenarios and use cases. The proposed network-assisted handover is described in detail in Chapter 4, and it is validated through testbed experiments in Chapter 5. Conclusions are drawn in Chapter 6.

The material of this thesis has been partially published in the following article:

Claudio Cicconetti, Francesco Galeassi, Raffaella Mambrini, "Network-Assisted Handover for Heterogeneous Wireless Networks", International Workshop on Seamless Wireless Mobility (SWM), co-located with IEEE Globecom 2010, Miami, FL, USA, 6-10 December, 2010.

Chapter 2

IEEE 802.21 Standard

2.1 Scope of the standard

This standard defines extensible IEEE 802 media access independent mechanisms which enable the optimization of handover between heterogeneous IEEE 802 networks and facilitate handover between 802 networks and cellular networks.

IEEE 802.21 supports handover for both mobile and stationary users. For mobile users, handover may occur for changes in the wireless link conditions and for a signal level degradation due to terminal mobility. For stationary users, handover may be required when the network environment or the user's requirements change.

Furthermore, the standard facilitates a variety of handover methods, including both hard handovers and soft handovers. A hard handover, also known as "*break-before-make*" handover, typically implies an abrupt switch between two Point Of Attachments (PoAs). Thus the connection to the source PoA is broken before the connection to the target PoA is made. Soft handover, usually called "*make-before-break*" handover, requires instead the establishment of a connection with the target PoA before the connection to the source PoA is broken. In soft handover, the MN remains briefly connected with two PoAs. Note, however, that depending on service requirements and application traffic patterns, hard handover may often go unnoticed. For example, web browsing and audio/video streaming with pre-buffering can be accommodated when handing over between different PoAs by employing mechanisms which allow transferring the node connection context from one PoA to another one.

Inside the IEEE 802.21, the term PoA (e.g., a wireless AP) describes the network side endpoint connected to the MN with a Layer 2 (L2) connection. Furthermore, the PoA associated with the MN is referred as the serving PoA, whereas any other PoA in the vicinity of the MN is a candidate PoA. Finally, the PoA which has been selected to become the next serving PoA is the target PoA.

Note that the standard does neither specifies policies for handover decisions nor determines whether the handover has to be mobile- or network-initiated. Its aim is to specify a framework in order to allow and facilitate such decisions.

The following subclauses give an overview of how different factors which affect handover are addressed within the IEEE 802.21.

2.1.1 Service continuity

Service continuity is defined as a continuation of the service during and after the handover while minimizing aspects suchs as data loss and duration of loss of connectivity during the handover process without requiring any user intervention. This standard specifies essential elements to maximize service continuity by providing seamless maintenance of active communications when the user changes its PoA, either wired or wireless.

2.1.2 Quality of service

This standard provides support for fulfilling application QoS requirements during handover. There can be a change in service quality as a consequence of the transistion between different networks due to varying capabilities and characteristics of access networks. For instance, lack of the required level of QoS support or low available capacity in a candidate access network may lead the higher layer entities to prevent a planned handover.

On the other hand, increasing delay, jitter or packet-loss rates in the currently serving access network may degrade the perceived QoS, triggering the mobility manager to start assessing the potential of candidate target access networks and subsequently initiate a handover. This standard includes mechanisms which support this aspect of QoS toward enabling seamless mobility, enabling the users to specify static QoS requirements and allowing the reception of dynamic information about the performance of the serving PoA and other PoAs in range. However, the standard does not specify any methods for collecting this dynamic information (such as packet loss and signal strength) at the link layer.

The Annex B of the standard [3] defines the parameters which are used to set the requirements and assess the performance of packet transfers between a source and its destination.

2.1.3 Network discovery and network selection

This standard includes elements that help in network discovery specifying the means by which such information can be obatined and made available to the MIH users. The network information includes information about link type, link identifier, link availability, link quality, etc.

Furthermore, it helps in network selection. A MN can select a PoA basing on various criteria such as required QoS, cost, user preferences or network operator policies. The information gathered are made available to the MIH users to enable effective network selection.

This way IEEE 802.21 can be used in conjunction with any mobility management protocols to facilitate handover. In fact, network discovery and network selection are outside the scope of these protocols (e.g., MIP and SIP).

Note that MIH users are abstractions of the functional entities which employ MIH service, that is, consumers of MIH services. A typical user of MIH services could be a mobility management application that would use these services to optimize handover.

2.1.4 Power management

The IEEE 802.21 allows the MN to discover different types of wireless networks (e.g., IEEE 802.11, IEEE 802.16 and 3G networks), avoiding powering up multiple radios and/or excessive scanning at the radios. Thus, this standard minimizes power consumed by mobile devices in the discovery of potential handover candidates. Specific power management mechanisms deployed are dependent on individual link-layer technologies and the potential power management benefits from this standard only extend to the discovery of wireless networks.

2.2 General overview

The IEEE 802.21 standard consists of the following elements:

- A framework which enables seamless transition of a MN between heterogeneous link-layer technologies.
- A new entity, called Media Independent Handover Function (MIHF), which realizes MIH functions.
- A media independent SAP, called MIH_SAP, that supplies the MIH users with services of the MIHF.
- A link-layer SAP, called MIH_LINK_SAP, for each link-layer technology, which provides an interface to the MIHF so as to control and monitor links.
- An abstract media dependent interface that provides transport service over the data plane on the local node, called MIH_NET_SAP.
- A new entity called MIH user, which is the functional entity that employs MIH services.

The MIH reference model is illustrated in Figure 2.1. The MIH_LINK_SAP is used to collect link information and control link behaviour during handover. For each existing network technology, amendments are needed to interact with the basic primitives defined in the IEEE 802.21 standard. Such work is being carried out for the IEEE 802.3, IEEE 802.11, IEEE 802.16, 3GPP and Third Generation Partnership Project 2 (3GPP2) in the respective task groups or technical committees.

On the other hand, it is envisaged that new technologies will incorporate suitable primitives for IEEE 802.21 natively.

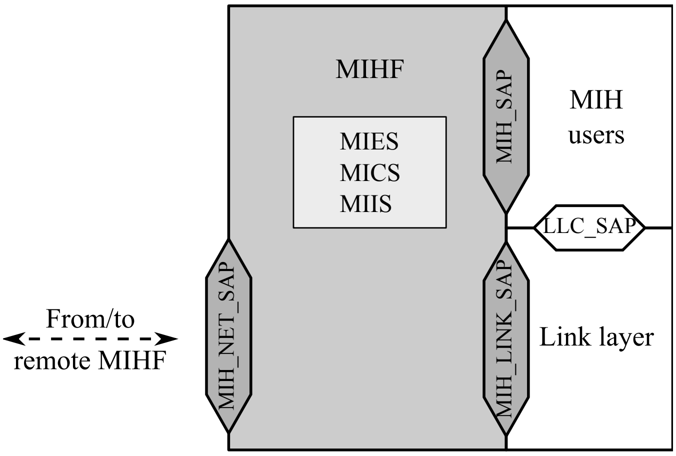

Figure 2.1: MIH reference model and SAPs

2.3 MIHF service

The MIHF provides the following services:

- A *Media Independent Event Service (MIES)*, which provides event classification, event filtering and event reporting corresponding to dynamic changes in link characteristics, link status, and link quality.
- A *Media Independent Command Service (MICS)*, which enables MIH users to manage and control link behaviour relevant to handover and mobility.
- A *Media Independent Information Service (MIIS)*, which defines a framework for acquiring, storing, and retrieving information useful for handover decisions within a geographical area. This information enables effective system access and effective handover decisions.

These services are managed and configured through service management primitives, as discussed in 2.3.4.

2.3.1 Media Independent Event Service

The Media Independent Event Service defines events which represent changes in dynamic link characteristics such as link status and link quality. Events may indicate changes in the state and the transmission behaviour of the physical, data link, and logical link layers or predict state changes of these layers. Due to the nature of the events, event notifications are generated asynchronously. As the upper layer gets notified about certain events it makes use of the command service to control the links.

A MIH user can subscribe to receive such notifications, both from the local lower layers and from remote entities, through the local MIHF. It is possible for multiple users to register to the same event, in which case the notification is sent to all the subscribers.

Events are broadly divided into two categories, Link Events and MIH Events. The firsts are defined as events which originate from event source entities below the MIHF and terminate at the MIHF. Entities generating Link Events include, but are not limited to, various IEEE 802-defined, 3GPP-defined, and 3GPP2-defined interfaces. The latter instead are defined as events which originate from within the MIHF and terminate at the MIH user, or they are Link Events that are propagated by the MIHF to the MIH users.

Events can be also classified as either local or remote. Local events are subscribed by the local MIHF and remain within a single node. Remote events are subscribed by a remote MIHF and are delivered over the network by MIH protocol messages.

2.3.1.1 Link events

The MIES supports the following categories of link events:

- *Medium Access Control (MAC) and Physical (PHY) State Change events:* These events correspond to changes in MAC and PHY state. For example, Link_Up and Link_Down are state change events.

- *Link Parameter events:* These events are due to changes in link-layer parameters. For example, Link_Parameters_Report is a link parameter event. This indication is used by the local MIHF to report the status of a set of link parameters, either local or remote.

- *Predictive events:* Predictive events convey the likelihood of a change in the link conditions in the near future based on past and present conditions. For example, decay in signal strength of a WLAN may indicate a loss of link connectivity in the near future.

- *Link Handover events:* These events inform upper layers about the occurrence of L2 handover or link switch if supported by the given media type[1].

- *Link Transmission Events:* These events indicate the link-layer transmission status (e.g., success or failure) of upper layer Protocol Data Units (PDUs). This information is used by the upper layers to improve buffer management so as to minimize the upper layer data loss due to a handover.

---

[1] The mechanism that triggers and executes a link-layer handover/switch (also referred as an L2 handover) is specified within the corresponding media-specific standard and out of scope of IEEE 802.21 standard.

For example, the occurrence of a hard handover of an MN from one PoA to another will result in the tear-down of the old link-layer connection between the MN and the source PoA and the establishment of a new link-layer connection between the MN and the target PoA. When this occurs, some upper layer PDUs still remain buffered at the old link, such as PDUs that had been queued at the old link but never been transmitted and PDUs that have been transmitted over the old link but never been acknowledged by the upper layer receiver before the link went down. When the handover succeed, these buffered PDUs will be discarded. As a result, unless the sender attempts to retransmit them over the new link connection, these PDUs will never reach the receiver.

Table 2.1: MIH events

MIH event	**(L)ocal, (R)emote**	**Description**
MIH_Link_Detected	L, R	Link of a new access network has been detected. This event is typically generated on the MN when the first PoA of an access network is detected.
MIH_Link_Up	L, R	L2 connection is established and link is available for use. This event is a discrete event.
MIH_Link_Down	L, R	L2 connection is broken and link is not available for use. This event is a discrete event.
MIH_Link_Parameters_Report	L, R	Link parameters have crossed prespecified thresholds. This notification can be also generated at a predefined regular interval to report the status of a set of link parameters (e.g., throughput).
MIH_Link_Going_Down	L, R	Link conditions are degrading and connection loss is imminent.
MIH_Link_Handover_Imminent	L, R	L2 handover is imminent based on changes in link conditions.
MIH_Link_Handover_Complete	L, R	L2 handover to a new PoA has been completed.
MIH_Link_PDU_Transmit_Status	L	Indicate transmission status of a PDU.

Table 2.1 show all the MIH Events defined in the standard. An MIH Event is marked as local only (L), remote only (R) or local and remote (L, R), indicating whether it can be subscribed by a local MIH user, a remote MIH user, or both, respectively. Note that for each MIH event exists a corresponding Link Event.

2.3.2 Media Independent Command Service

The Media Independent Command Service provides a set of commands to configure, control and retrieve information from the lower layers. If an MIHF supports the command service, all MIH commands are mandatory in nature; when an MIHF receive a command, it is always expected to execute the command. Typical commands are used to configure network devices and scan available networks. They are classified into two categories (i.e., Link Commands and MIH Commands) and can be invoked either locally or remotely by the MIH users or by the MIHF itself.

Link Commands originate from the MIHF and are directed to the lower layers. They mainly control the behaviour of the lower layer entities and are only local. Whenever possible, existing media-specific link commands must be used for interaction with specific access networks. Although originating from the MIHF, these commands are executed on behalf of the MIH users. The standard defines six Link Commands:

- Link_Capability_discover
- Link_Event_Subscribe
- Link_Event_Unsubscribe
- Link_Get_Parameters
- Link_Configure_Thresholds
- Link_Action

MIH Commands are instead generated by the MIH users and sent to the MIHF. They can be local or remote.

Local commands propagate from the MIH users to the MIHF and then from the MIHF to the lower layers. Remote commands are carried by MIH protocol messages and can be propagated from the local MIHF to a remote MIHF. Table 2.2 shows the MIH commands defined in the standard.

The MICS facilitates both mobile-initiated and network-initiated handovers defining different commands depending on the origination and the destination point, as shown in Table 2.3. These commands allow both the MN and the network to initiate handover and exchange information about available networks, negotiating the best available PoA.

In Annex C of the IEEE 802.21 standard [3] are described some example procedures for both mobile-initiated and network-initiated handover.

2.3.2.1 Mobile-initiated handovers

The network selection policy function resides on the the MN which initiates the handover using the set of MIH_MN_HO_*** commands. The MN can use these commands to obtain the list of available candidate networks, reserve any required resources at the candidate target network and indicate the status of a handover operation to the MIHF in the network.

2.3.2.2 Network-initiated handovers

In this case, the network selection policy function resides on the network which initiates the handovers using the set of MIH_Net_HO_*** commands in conjunction with MIH_N2N_HO_*** commands. The network entity can use these commands to obtain the list of resources currently used by the MN and to reserve required resources at the candidate target network. Finally, it can command the MN to performe a handover to a specific network.

Table 2.2: MIH commands

MIH command	**(L)ocal, (R)emote**	**Description**
MIH_Link_Get_Parameters	L, R	Get the status of a link.
MIH_Link_Configure_Thresholds	L, R	Configure link parameter thresholds.
MIH_Link_Actions	L, R	Control the behaviour of a set of links.
MIH_Net_HO_Candidate_Query	R	Network initiates handover and sends a list of suggested PoAs to the MN.
MIH_MN_HO_Candidate_Query	R	Handover is initiated by the MN, which uses this command to query and obtain information about possible candidate networks.
MIH_N2N_HO_Query_Resources	R	This command is sent by the serving MIHF entity to the target MIHF entity to allow for resource query.
MIH_MN_HO_Commit	R	Command used by MN to notify the serving network of the decided target network information.
MIH_Net_HO_Commit	R	Command used by the network to notify the MN of the decided target network information.
MIH_N2N_HO_Commit	R	Command used by a serving network to inform a target network that an MN is about to move toward that network, initiate context transfer, and perform handover preparation.
MIH_MN_HO_Complete	R	Notification from MIHF of the MN to the target or source MIHF indicating the status of handover completion.
MIH_N2N_HO_Complete	R	Notification from either source or target MIHF to a peer MIHF indicating the status of the handover completion.

Table 2.3: Naming convention for MIH handover command primitives

Primitive name prefix	Originating point	Destination point
MIH_MN_HO_***	MN	Network
MIH_Net_HO_***	Network	MN
MIH_N2N_HO_***	Network	Network

2.3.3 Media Independent Information Service

The Media Independent Information Service defines a framework for discovering and obtaining network information within a geographical area to facilitate network selection and handover. The goal of the MIIS is to supply either the MNs or the network with a global view of all heterogeneus networks in the area so as to facilitate seamless roaming across these netwoks.

Support for many Information Elements (IEs) is included to encompass the different types of mobility and supported technologies. The MIIS provides the capability for obtaining information about lower layers such as neighbor maps and other link-layer parameters, as well as information about available higher layer services (e.g., Internet connectivity), allowing this information to be accessed from any single network. For example, toward an IEEE 802.11 interface, the MN can get information not only about other IEEE 802 based networks in a particular area but also about 3GPP and 3GPP2 networks. The MN can use its currently active interface to inquire about other available access networks in a geographical region, thus removing the need for scanning. Furthermore, the MN can optimally configure the radio for connecting to selected access network. The MIIS enables these functionalities across available access networks by providing a uniform way to retrieve heterogeneous network information in a geographical area.

The information provided by the MICS is dynamic, referring to link parameters (e.g., signal strength and link speed), whereas the MIIS provides less dynamic information. Anyway, both information could be used in combination either by the MN or the network so as to facilitate the handover.

2.3.3.1 Information elements

Information Elements are classified into the following three groups:

- *General information and access network specific information:* these IEs give a general overview of the different networks within a geographical area. For example, they may contain the list of available networks and their associated operators, roaming agreements between different operators, the cost of the connection, network security and QoS capabilities.
- *PoA specific information:* these IEs provide information about available PoAs in each access networks in the area. These IEs contain PoA address-

ing information, PoA location, data rates supported, the type of PHY and MAC layers and any channel parameters to optimize link-layer connectivity, as well as higher layer services and individual capabilities of each PoAs. With careful planning and by taking advantage of this information, mobile nodes may be able to reduce the number of handovers and optimize the use of network resources.

- Other information that is access network specific, service specific or vendor/network specific.

Table 2.4 shows some IEs defined in the standard.

Table 2.4: Example of Information Elements

Information Element	**Description**
General Information Elements	
IE_NETWORK_TYPE	Link types of the access networks that are available in a given geographical area.
IE_OPERATOR_ID	The operator identifier of the access network/core network.
Access network specific Information Elements	
IE_NETWORK_ID	Identifier of the access network.
IE_COST	Indication of the cost for service or network usage.
IE_NETWORK_QOS	QoS characteristics of the link layer.
PoA-specific Information Elements	
IE_POA_LINK_ADDR	Link layer address of the PoA.
IE_POA_IP_ADDR	IP Address of the PoA.
IE_POA_CHANNEL_RANGE	Channel range supported by the PoA.
Other Information Elements	
Vendor-specific IEs	Vendor-specific services.

2.3.3.2 Information elements containers

Three IE containers are defined in the standard:

- *IE_CONTAINER_LIST_OF_NETWORKS*, which provides a list of neighboring IE_CONTAINER_NETWORK, containing information that depicts a list of heterogeneous neighboring access networks for a given geographical area and optionally one or more vendor-specific IEs. If more than one access network is provided, they should be priorized in the order of preference.

- *IE_CONTAINER_NETWORK*, which contains the information decipting an access network and a list of IE_CONTAINER_POA.

- *IE_CONTAINER_POA*, which contains information that decipts a PoA and otionally one or more vendor-specific IEs.

2.3.3.3 Information elements representation

The IEs can be encoded either in binary form, using Type-Length-Value (TLV) structure, or with a Resource Description Framework (RDF)[2] representation, similar to the Extensible Markup Language (XML). Data retrieval is done by means of TLV or SPARQL[3] queries, for each respective encoding method. Using the RDF framework, information provided by the MIIS conforms to a specific structure and semantic defined by the RDF schemas. RDF provides support for efficiently responding to complex queries. For example, using the RDF schema, the MIIS can be used to quickly and efficiently identify nearby networks that meet a complex set of specific criteria. Relying on TLV exchanges for the same queries may result in a greater number of message exchanges, which require more bandwidth and introduce greater handover latency.

The RDF schema definition consists of two part: the basic and the extended schema. The basic schema is provided in the Annex H of the standard [3], while the extended schema contain user- or vendor-specific IEs. An example of a vendor-specific extension is illustrated in the Annex I of the standard [3]. Network and other informations (e.g., RDF schemas) can be stored in a network element referred to as MIIS server. Two primitives have been defined in the standard either to get information from the MIIS server (MIH_Get_Information) or to enable the MIIS server to push information to MNs (MIH_Push_Information), so as to update policy informations after a successful registration. The RDF schemas can be pre-provisioned to the MNs or can be obtained by means of the two primitives described above.

2.3.4 Service management

In order to properly function, MIH entities need to be configured through the following service management functions defined in the IEEE 802.21 standard:

- *MIH capability discover*, a procedure used by an MIH user to discover local or remote MIHF capabilities in terms of MIH services (Event Service, Command Service, and Information Service). Depending on the information obtained from this procedure, the local MIHF can determine which peers it should register with. The MIH capability discover function can use either the MIH protocol, described in 2.6, at L2 and L3, or media specific L2 broadcast message. For example, an MIHF can listen to media specific broadcast message, such as IEEE 802.11 beacons or media independent L2

---

[2] http://www.w3.org/RDF/.

[3] http://www.w3.org/TR/rdf-sparql-query/.

MIH_Capability_Discover broadcast messages, beacuse an MIHF entity residing in the network may periodically announce its presence and capabilities. An MIHF can also send MIH_Capability_Discover.request using broadcast or unicast messages so as to discover peers in a solicited way. For instance, an MIHF can send a unicast request for obtaining the capabilities of a specific IEEE 802.21 network entity. In this case, only the addressed network entity should respond to this request.

- *MIH registration*, a procedure used by an MIH user to register its local MIHF with a peer in order to establish a MIH session. After a successful registration, the two peers can request access to specific MIH services. For example, MIH registration can be used by an MN to declare its presence to a selected MIH Point Of Service (PoS) or to subscribe to a particular set of events by means of the MIH event subscription. Inside the standard, the term PoS is referred to the peer located in the network node the MN communicates with. Note that MIH registration is not necessary for obtaining some level of support from a peer. However, by registering and authenticating, peers typically will get access to much more extensive information. Although the MIHF residing on the MN may be able to access information from the network-side peers without registering, the available information may be only a subset of that provided after authentication.
- *MIH event subscription*, a mechanism that allows an MIH user to subscribe or unsubscribe to a particular set of events that originates from a local or remote MIHF. Event subscription from a peer requires registration. The subscription contains only the list of events the MIH user is interested in. Note that event sources may not support the events that the subscriber is interested in subscribing to. Each subscription request is matched by a confirmation message from the event source indicating the events approved for the subscription.

Table 2.5: Service management primitives

Service management primitive	**(L)ocal, (R)emote**	**Description**
MIH_Capability_Discover	L, R	Discover the capabilities of a local or remote MIHF.
MIH_Register	R	Register with a remote MIHF.
MIH_DeRegister	R	Deregister from a remote MIHF.
MIH_Event_Subscribe	L, R	Subscribe for one or more MIH events with a local or remote MIHF.
MIH_Event_Unsubscribe	L, R	Unsubscribe for one or more MIH events from a local or remote MIHF.

2.4 Service Access Points

Service Access Points are used to enable the communication between the MIHF and other layers. Every SAP defines a certain number of primitives that describe the communication with the services in the MIHF. Note that the standard does not mandate a specific programming language for representing these primitives. The specification of the MIHF includes the definition of SAPs that are media independent and recommendations to define or extend other SAPs which are media dependent. Figure 2.2 shows the relationship between the different SAPs defined in the standard.

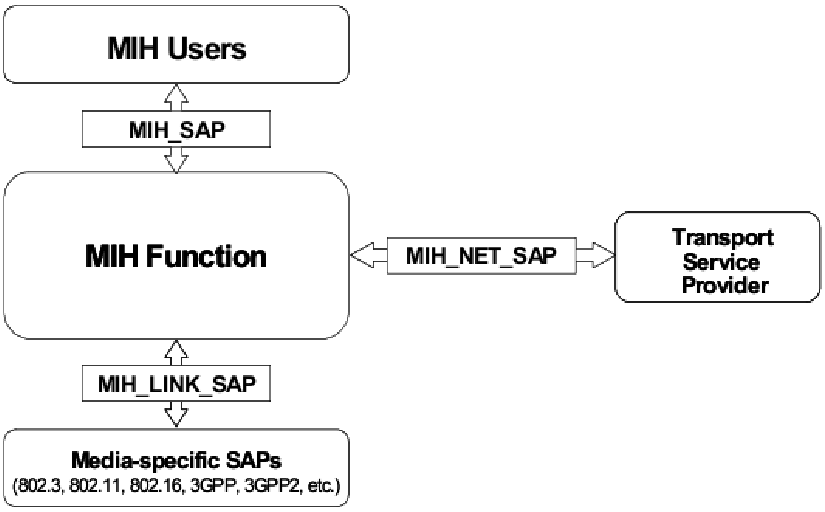

Figure 2.2: Relationship among different SAPs

The *MIH_SAP* specifies a media independent interface between the MIHF and the upper layers. Upper layers need to subscribe with the MIHF as users to receive MIHF generated events and link layer events that originate at layers below the MIHF but are passed to MIH users through the MIHF. MIH users directly send commands to the local MIHF using the service primitives of the MIH_SAP.

The *MIH_LINK_SAP* specifies an abstract media dependent interface, which allows the MIHF to use services from the lower layers. All inputs (including the events) from the lower layers of the mobility-management protocol stack into the MIHF are provided through existing media-specific SAPs such as MAC, PHY and Logical Link Control (LLC) SAPs. Link Commands generated by the MIHF to control the PHY and MAC layers during the handover are part of the media-specific MAC/PHY SAPs and are already defined elsewhere.

Communication between two MIHF peers relies on MIH protocol messages. This service is enabled by the *MIH_NET_SAP*, an abstract media dependent interface, which provides transport services over the data plane on the local node. For L2, this SAP uses the primitives provided by MIH_LINK_SAP.

The following primitives associated with data transfers are defined in the standard:

- *MIH_TP_Data.request*, which is used by an MIHF to request the delivery of an MIH PDU.
- *MIH_TP_Data.indication*, which is used by the transport service provider to indicate the arrival of an MIH PDU.
- *MIH_TP_Data.confirm*, which is used to acknowledge the successful transfer of an MIH PDU.

2.5 MIHF communication model

The MIHF communication model specifies different MIHF roles and their communication relationships, such as supported transport mechanisms and service classes. The assigned MIHF roles depend on their location in the network. For example, an MIHF on a mobile node can directly communicate with network-side entities (i.e., the MIH PoSs) using L2 or L3 communication. MIH PoSs may include the serving PoA or candidate PoAs. Network-side MIHF entities can communicate with each other at L3 or above using the MIH protocol. Figure 2.3 shows an example scenario of the communication model.

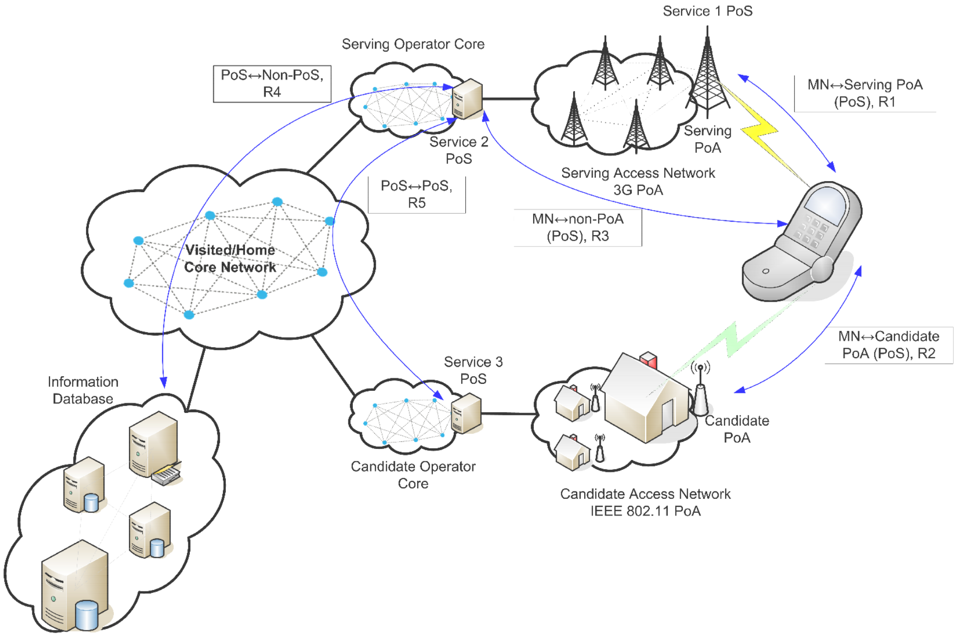

Figure 2.3: MIHF communication model

2.6 MIH protocol

MIHF peers communicate with each other using MIH protocol messages. The MIH protocol defines the header format, content and encoding for a set of protocol messages and the mechanisms that support the delivery of these messages. They are composed of an eight byte header part and a TLV encoded payload part. For internal communication, no particular encoding is dictated. For the maximum flexibility, the communication protocol between MIHF entities is specified by the standard for both L2 and L3: L2 transport is allowed with the EtherType value set to that for MIH Protocol; L3 transport is supported for Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and Stream Control Transmission Protocol (SCTP).

An acknowledgment service can be enabled to add reliability to message exchanges in case where the transport method adopted does not already provide this. The sender MIHF can set the Acknowledgement (ACK)-Req field so as to instruct the receiver to return an acknowledgement message with ACK-Rsp bit set. The MIH Message Identifier (ID) and Transaction ID must be the same in the request message and its acknowledgement. An acknowledgement message may carry no payload. Note that despite employing these two ID fields, the MIH protocol does not specify any further mechanisms for reliable authentication.

Furthermore, the MIH protocol defines an MIH fragmentation mechanism, which enables messages fragmentation. An MIH message is fragmented only when the MIH message is natively sent over an L2 medium, such as Ethernet. When MIH messages are sent using an L3 or higher layer transport, L3 must take care of any fragmentation issue.

2.6.1 MIH protocol transaction

The MIH protocol defines a message exchange between two MIHF entities to support remote MIHF services. An MIH transaction is identified by a sequence of messages with the same Transaction ID exchanged between two remote MIHF entities. At a given moment, an MIH node shall have no more than one transaction pending for each direction with a certain MIH peer. For instance, the MIH node shall wait until any pending outgoing transaction is completed before it can create another outgoing transaction for the same peer.

2.6.2 MIH protocol identifiers

The following identifiers are used in MIH protocol messages:

- *MIHF Identifier*, which is an identifier that uniquely identify an MIHF entity and is assigned to the MIHF during its configuration process. The MIHF ID shall be invariant and could be, for example, a Fully Qualified Domain Name (FQDN) or a Network Access Identifier (NAI). All MIH protocol messages contain the source and the destination MIHF ID in the payload part, enabling the MIH protocol to be transport agnostic. A multicast MIHF ID (i.e., a zero length MIHF ID) can be used when the destination MIHF ID is not known to a source MIHF.

- *Transaction Identifier*, which is an identifier that is used to match a request message with its corresponding response message. This identifier is also required to match each message with its corresponding acknowledgement. It is created at the node initiating the transaction and it is carried over in the header of the MIH protocol frame.

2.6.3 MIH protocol frame format

In MIH protocol messages, all TLV definitions are always aligned on an octet boundary and hence no padding is required. An MIH protocol payload carries a source MIHF Identifier TLV and a Destination MIHF Identifier TLV followed by MIH service specific TLVs. Figure 2.4 shows the format of the MIH protocol frame.

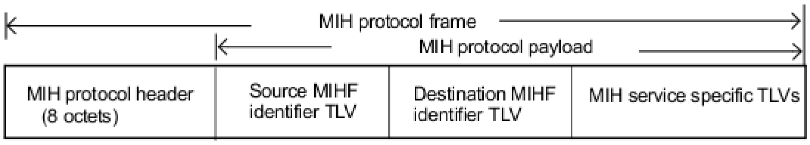

Figure 2.4: MIH protocol general frame format

The MIH protocol header carried the essential information which is present in every frame and is used for parsing and analyzing the MIH protocol frame, as shown in Figure 2.5.

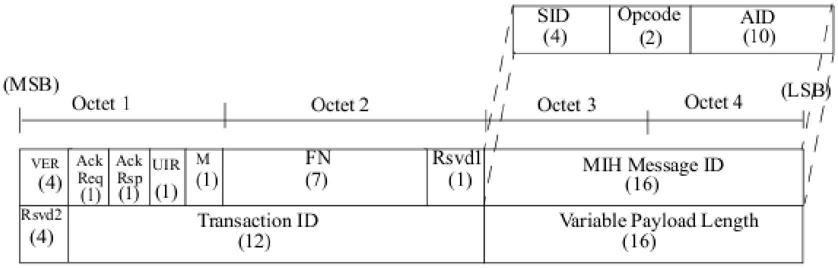

Figure 2.5: MIH protocol header format

The *Version* field in the MIH frame header specifies the version of the MIH protocol used while the two *ACK* fields are for acknowledgement purposes. The *Unauthenticated Information Request (UIR)* flag indicates that the response message may be sent with a limited length because of the nature of unauthenticated message exchange.

Table 2.6: Description of the MIH protocol header fields

Field name	**Size (bits)**	**Description**
Version	4	This field is used to specify the version of MIH protocol used. An MIH node that receives an MIH message with a higher version number than it supports will discard the frame without indication to the sending MIH node.
ACK-Req	1	This field is used for requesting an acknowledgement for the message.
ACK-Rsp	1	This field is used for responding to the request for an acknowledgement for the message.
UIR	1	This field is used by the MIIS to indicate if the protocol message is sent in pre-authentication/pre- association state so that the length of the response message can be limited.
More fragment (M)	1	This field is used for indicating that the message is a fragment to be followed by another fragment.
Fragment number (FN)	7	This field is used for representing the sequence number of a fragment.
Reserved1	1	This field is intentionally kept reserved.
MIH message ID (MID)	16	Combination of the following 3 fields.
– Service Identifier (SID)	4	Identifies the different MIH services.
– Operaction Code (Opcode)	2	Type of operation to be performed with respect to the SID.
– Action Identifier (AID)	10	This indicates the action to be taken with regard to the SID.
Reserved2	1	This field is intentionally kept reserved.
Transaction ID	12	This field is used for matching Request and Response, as well as matching Request, Response and Indication to an ACK.
Variable payload length	16	Indicates the total length of the variable payload embedded in this MIH protocol frame. The length of the MIH protocol header is NOT included.

Recall that, when an MIHF issues requests without registering first with its peer, it may receive less information than if it had registered earlier. If this flag is set, then the information included in the response message may not reflect the complete information available to registered peers. The *More Fragments (M)* and *Fragment Number (FN)* fields are used in messages fragmentation.

The *MIH Message ID* field comprises three subfields. The *Service Identifier (SID)* field indicates the MIHF service class (MIES, MICS, MIIS or Service Management) which this message belongs to. The *Operation code (Opcode)* specifies whether the message is a request, response or indication. The *Action Identifier (AID)* is related with and scoped by the SID. For instance, if the SID indicates MIES, AID points to the actual event type. The *Variable Load Length* field contains the total length of the variable TLV-encoded payload carried by this message frame.

The MIH protocol messages use the *Transaction ID* and *MIHF ID* fields as identifiers, but only the former is included in the header. The *Transaction ID* field is used to match each request, response or indication message with its acknowledgement. Table 2.6 shows the description of the header fields.

2.7 Specific technologies amendments

The MIHF aggregates disparate interfaces with respective media dependent lower layer instances (i.e., media dependent SAPs) into a single interface for the MIH users (i.e., MIH_SAP), reducing the inter-media differences to the extent possible. The MIHF features media dependent interfaces with IEEE 802 link-layer technologies (i.e., IEEE 802.3, IEEE 802.11 and IEEE 802.16) and cellular technologies (i.e., 3GPP). The MIHF for the most part uses existing primitives and functionality provided by different access technology standards. Amendments to existing standards are recommended only when necessary to fulfill the MIHF capabilities.

2.7.1 IEEE 802.11 amendments

IEEE 802.11u is a TG, which was chartered to allow devices to inter-work with external networks, as typically found in hotspots. It assists the advertising and the connection to remote services and it intends to provide information to the MN about the external networks prior to association. One of the goal of this group is to enable the integration of the IEEE 802.11 standard with the IEEE 802.21 framework. To support MIH services, appropriate primitives are added in IEEE 802.11, such as the definition of the *MAC State Generic Convergence Function - SAP (MSGCF-SAP)*, which includes primitives related to the following:

- System configuration
- Link state change notifications/triggers
- MIH frame transport through control or management frames

The MIHF uses the MSGCF-SAP for interfacing with the link-layer of IEEE 802.11 networks. The MIH_LINK_SAP defines additional primitives that map to MSGCF-SAP.

2.7.2 WiMAX amendments

IEEE 802.16g is a TG, which was chartered to provide conformant 802.16 equipment with procedures and services, to enable interoperable and efficient management of network resources, mobility, and spectrum and to standardize management plane behaviour in 802.16 fixed and mobile devices. The *Control-SAP (C-SAP)* and the *Management-SAP (M-SAP)* have been extended to support MIH related primitives. The MIHF uses the C-SAP and the M-SAP for interfacing with the Control and Management planes of the IEEE 802.16 network. The C-SAP includes primitives related to the following:

- Handovers
- Subscriber and session management
- Radio resource management
- Media independent function services

The M-SAP includes primitives related to the following:

- System configuration
- Monitoring statistics
- Notifications triggers
- Multi-mode interface management

2.8 Summary and open challenges

An important challenge facing the IEEE 802.21 is the unification of all media-specific technologies under one abstract interface (i.e., MIH_LINK_SAP). This approach may be difficult to realize in practice within a short period of time due to the large number of technology-specific standards within and outside the IEEE 802 systems, which must be extended to conform to the MIH_LINK_SAP. In certain technologies, media-specific primitives may be already available and is only required the correctly mapping of them to MIH_LINK_SAP primitives. Other technologies, however, may require extensions to media-specific primitives.

The IEEE 802.21 WG has established liaisons with the IEEE 802.11 TG "u" and the IEEE 802.16 TG "g". To support MIH services, appropriate primitives will be added to the IEEE 802.11 and the IEEE 802.16 standards. Proposals have been made to the 3GPP Standards Association to incorporate MIH services so as to support handover between WiMAX, 3GPP and LTE networks. Although the inclusion of these proposals into the Release-8 specification for the 3GPP was not accepted, MIH remains an attractive technology for enhancing IP mobility across heterogeneous accesses for future 3G releases. Furthermore, the Internet Engineering Task Force (IETF) Mobility for IP Performance, Signaling, and Handoff Optimization (MIPSHOP) WG is specifying an higher layer transport for the MIH protocol and mechanisms to discover MIHF peers.

Another challenge to the widespread adoption of the IEEE 802.21 is the lack of a conformance statement detailing the mandatory set of primitives and primitive sequences required to realize a particular use case. Such a statement would provide a method to verify that IEEE 802.21-based equipments conform to the standard and assurances to the community that equipments from different vendors will interoperate with each other.

The IEEE 802.21 standard must also address a number of additional features to ensure acceptance. Some features are better left to individual companies to be addressed, giving them an opportunity to distinguish themselves in the marketplace. However, if a required set of common hooks and interfaces is not in place and interoperability is not ensured, industry adoption will be difficult. Important features to the deployment of MIH services, such as MIIS provisioning, MIH security and multi-radio power management are not fully addressed in the specification yet. MIIS provisioning deals with issues concerning how information is populated to and stored in the MIIS server. MIH security includes mechanisms to protect MIH protocol messages based on mutually authenticating MIH entities. Because operating multiple radios can be a significant drain on the battery of a device, mechanisms must be in place to facilitate better power management for multi-radio devices.

In summary, the success of this standard depends not only upon the activities within the IEEE 802.21 WG, but also upon the acceptance of this technology by other standards and industry forums. To achieve acceptance and wide deployment in the future, additional specifications describing use case scenarios with requirements, features and required extensions to media-specific technologies are required. This means continued effort on the part of the IEEE 802.21 WG, as well as close collaboration and liaison with other standard organizations.

Chapter 3

Problem statement

3.1 Definition of the problem

The rapid increase of overlapping access networks has led to the need for MNs capable of roaming seamlessly across heterogeneous access technologies such as IEEE 802.11, WiMAX and Universal Mobile Telecommunications System (UMTS) as well as between wired networks (e.g., Digital Subscriber Line (DSL) and cable). Supporting seamless roaming between heterogeneous networks is a challenging task since each access network may have different mobility, QoS and security requirements. Moreover, interactive applications such as VoIP and streaming media have stringent performance requirements on end-to-end delay and packet loss. The handover process stresses these performance limits by introducing delays due to discovery, configuration, authentication and binding update procedures associated with a mobility event.

The overall handover delay can be attributed to operational delays at all layers of the protocol stack including L2, L3 and the application layer. Performance can also be tied to the specific access networks and protocols that are used for network access. For example, configuring a Point-to-Point Protocol (PPP) interface in a Wide Area Network (WAN) environment takes more time than configuring an interface using Dynamic Host Configuration Protocol (DHCP) in a Local Area Network (LAN) environment.

Access network-specific authentication and authorization protocols may introduce additional delays. It is observed that traditional non-optimized handover takes up to 4 sec. delay during inter-LAN movement. Thus in a typical deployment scenario, several hundred packets may be lost during the handover. Besides, it may take up to 15 sec. to complete authentication and connection establishment procedures if the neighboring network is either a UMTS or a General Packet Radio Service (GPRS) network. Movement between two different administrative domains poses additional challenges since a mobile will need to re-establish authentication and authorization in the new domain. L2 handoff delay is more relevant when an authentication process is involved to obtain L2 connectivity.

Moreover, the latency introduced due to scanning and authentication at L2 is not acceptable for real time communications. For example, in IEEE 802.11 based wireless networks, the IEEE 802.11i security mechanism performs a new set of exchanges with the authenticator in the target AP in order to initiate an Extensible Authentication Protocol (EAP) exchange to an authentication server. Following a successful authentication, a 4-way handshake with the wireless station derives a new set of session keys for use in data communications. This process can significantly prolong the handover event and calls for improved latency performance in L2 security mechanisms related to handover.

Other problems are related to network discovery and selection. Each wireless technology in integrated networks has its own characteristics that complement others. For instance, WLANs support higher data rates than other wireless access technologies, but it provides small coverage areas. In contrast, WiMAX and 3G networks cover relatively large areas, but they provide smaller data rates than IEEE 802.11 networks.

Due to such distinctive characteristics, the integration of WLANs with other wide area access networks is one of the main problem in such heterogeneous networks. One solution could be always to leave on the IEEE 802.11 interface in order to attemp to connect to WiFi APs when they are available. This would lead to an inefficient use of energy since WLANs are not always accessible due to their small service coverage. For efficient power management of MNs which are battery-powered, keeping IEEE 802.11 interface turned on should be avoided. A possible solution to reduce energy consumption is to turn on the WLAN interface for a certain predefined time interval and try to associate with the WLAN AP. This is quite simple but a more efficient solution is described in Chapter 4.

Furthermore, in urban settings, the number of available access network can be very high. MNs are not capable of recognizing the PoAs they find so they can not know if they have either the proper security credentials to access the network or even if the network provides Internet access, and as result they do not know whether to attempt association. They can not go on attempting association with available PoAs until they make connection because every attempt depletes the battery life. Furthermore, network selection can be too complicated for non-technical users, because they may not be able to recognize PoAs by the network name (e.g., Service Set Identification (SSID) for WiFi networks).

3.2 Use cases

Nowadays there is an increase in the number of mobile hosts that often migrate during active data transfer and expect the network to manage these handovers with minimum disturbance to on-going data sessions. The working sessions have to be kept active when the users leave an access provider for a new one or when for a short time no provider is available. Therefore there is a need to find solutions for seamless handover among heterogeneous networks that support a smooth switch between different access technologies. Such solutions are useful to anyone who uses Internet in mobility and/or in places where the wireless connection is unstable.

For example, students and researches could move within their campus handing off from one WiFi hotspot to another, without loosing the connection. Professionals, consultants, brokers, insurance agents, lawyers, could work on a train or in a taxi, send and receive e-mails, remotely access documents in their office without any user intervention. Employees could move from one floor to another of an enterprise building, remaining connected to the WiFi intranet without any need to re-authenthicate.

Furthermore, a framework that enables seamless handover can be useful in many other cases, from remote healthcare monitoring to Vehicular Communication (VC). Here is a list of real scenarios chosen to underline the need for smooth handover.

Remote healthcare monitor A remote healthcare monitoring system is a comprehensive monitoring system to capture, transmit and distribute vital health data to doctors, patients and family. It monitors patients with chronic or long-term illnesses such as diabetes or cardiovascular disease, and patients discharged after an operation or serious medical crises, such as stroke victims. It can acquire vital information about a patient who lives far away from medical support and it can alert medical staff if there is a dangerous change in the patient's status. This means real-time monitoring without high staff or capital costs, very critical matters since the quality and cost of healthcare continue to be significant issues for every nation in the EU and perhaps in the world. Furthermore, the healthcare system can in some case be enabled to distribute medicine (e.g., insulina) at a particular time of the day, for example to disabled patients. The use of seamless handover procedures allow the monitoring system both to switch between various PoAs in a seamless way when the patient goes out and to automate all the procedures so that the handover is completely transparent to the final user. This way, also a user that has no expertise could use these instruments. One of the main problem of this architecture is to guarantee service continuity, regardless of patient position. For example, if a patient has to go out to do some shopping, the monitoring system which is carried by the patient must change the current PoA in a seamless way in order to guarantee service continuity. This is vital for such real-time applications where the breaking of the link can be very dangerous, especially if the system is composed of a monitor and an actuator. Figure 3.1 shows the Mobile-CliniQ [1], a Symbian-based cellphone application software, which allows the monitoring of various parameters and communicates with vital sign monitoring devices via a Bluetooth interface. Data is sent via GPRS to a medical monitoring center.

---

[1] http://www.aerotel.com/.

Vehicle communication Vehicle-to-Infrastructure (V2I) communication can enable a range of applications to enhance transport safety and efficiency, as well as infotainment. For example, it can be used to send warnings about environmental hazards (e.g., ice on the road), traffic and road conditions (e.g., emergency breaking, congestion or road works) and local (e.g., tourist) information. The main problem with this architecture is that the link is continuously changing due to the mobility of the vehicle, the speed of the which might be high, thus leading to many change of the current PoA. In addition, communication inside the car and to other cars (i.e., Vehicle-to-Vehicle (V2V)) have to be managed. Furthermore, automotive service workers may like to connect to the on board diagnostic unit of a car using a wireless link in order to read out fault codes or to run a detailed diagnosis.

Figure 3.1: Remote healthcare monitor

IPTV IPTV is a system through which Internet television services are delivered using the architecture and networking methods of the IP suite over a packet-switched network infrastructure, instead of being delivered through traditional radio frequency broadcast, satellite signal or cable television formats. This helps consumers watch TV on a Personal Computer (PC) or on other compatible forms of media such as a mobile telephone or a gaming console. IPTV services may be classified into three main groups: live television, time-shifted programming and Video-on-Demand (VOD). In commercial environments IPTV is widely deployed for distribution of live TV, video playout channels and VOD material across LAN or WAN IP network infrastructures, with a controlled QoS. The number of global IPTV subscribers is expected to grow from 28 million in 2009 to 83 million in 2013. This fact joined with the fast spreading of cell phone supporting WiFi and 3G wireless might lead to a profileferation of IPTV services not only related to the home environment. Users can watch TV either while on the bus going to work or while traveling by subway. The main problem is to minimize both the latency of video transmissions and the latency in response to a user request, especially when the user is making handover. Video buffering can limit this problem.

VoIP communication Internet telephony refers to communications services (e.g., voice, facsimile and/or voice-messaging applications) that are transported via the Internet, rather than the PSTN. The basic steps involved in originating an Internet telephone call are conversion of the analog voice signal to digital format and compression/translation of the signal into IP packets for transmission over the Internet; the process is reversed at the receiving end. Because of the bandwidth efficiency and low costs that VoIP technology can provide, businesses are gradually beginning to migrate from traditional copper-wire telephone systems to VoIP systems to reduce their monthly phone costs. VoIP solutions aimed at businesses have evolved into unified communications services that treat all communications (e.g., phone calls, faxes and e-mail), as discrete units that can all be delivered via any means and to any handset, including cell phones. Furthermore, the spreading of dual mode cell phones, which allow for seamless handover between a cellular network and a WiFi network, are leading to an increase of VoIP popularity. Thus, traditional mobile calls have to deal with VoIP, which due to lower communication and infrastructure costs, is becoming more and more popular. Handover procedures should be executed in such a way that any perceptible interruption to the conversation will be minimized.

Online music distribution The distribution of online music has played a key role in driving the digital multimedia entertainment business forward to the pervasive point we know today [23]. Nowadays it is possible either to buy music, to sell your own music and to listen to the radio online. Many Internet radio services are associated with a corresponding traditional radio station or radio network and usually offer a podcasting service, to enable users to download online contents, such as video music and daily programs. As an example, a program like iTunes enables the users to purchase and download music, music videos, television shows, podcasts and to rent films from the online iTunes Store. Furthermore, the advent of high-speed wireless networks, together with the spread of smartphones, is making it possible to supply these services to mobile and nomadic users. For instance, a mobile user could download a song without any service interruption, while travelling in a bus from the university to the home, using available 3G networks. When he arrives home, the MN automatically switches to the avilable WLAN connection and completes the download without any user intervention. The main problem is, as previously described, that the link is continuously changing due to the mobility of the users.

Below are described some example scenarios where the IEEE 802.21 standard can improve the user's experience by facilitating seamless handover procedures both intra-technology and inter-technology in order to achieve the best Quality Of Experience (QoE) for the users.

3.3 Example scenarios

The first scenario represents an example of horizontal handover between WiFi networks with a multi-interface MN. The second scenario illustrates the previous one, in a case where the MN has only one interface. The third scenario describes an example of vertical handover between WiFi and 3G networks. Finally, the last scenario represents an example of vertical handover between WiFi and WiMAX networks with a multi-mode MN.

3.3.1 Handover between WiFi networks with a multi-interface MN

Figure 3.2 shows a V2I communication scenario. The MN is represented by a car on the motorway equipped with a vehicle communication gateway, which acts as a central communication device, with two IEEE 802.11 interfaces. The gateway can also manage in-vehicle communication through a Bluetooth interface, enabling the user to listen to the music from an online music repository or to get live traffic and travel information. On the motorway there are overlapping IEEE 802.11 PoAs that enable V2I communication. To enable a seamless access to the infrastructure, the gateway autonomously switches among the best available PoAs at the current time and location. The best available network is selected from a list of all possible networks according to a user or network policy. The policy can consider user preferences, operator contracts, QoS requirements of the service in use and finally security requirements.

MIH services can be integrated in the network entities (e.g., WiFi PoA) and in the MN to enable seamless handover. With a multi-interface MN we can use a soft handover. For example, while a MN is associated with the serving PoA, it can query the MIIS server to obtain the next available WiFi network information through the associated interface without activating or directly scanning through the unused WiFi interfaces. This information can be used to configure and activate the not associated interface when the MN is leaving the coverage area of the serving PoA. This way, both the interfaces can be up and associated with different PoAs for a brief time. The MIES triggers a new handover, while the MICS commands allow either the MN or the network to begin the handover process. This way, both handover latency and the energy consumption of the mobile device are reduced and service continuity is guaranteed.

Some possible uses of this architecture would be to support the roadside assistants with advanced remote diagnosis services, to enable traffic control applications, as well as to advise the drivers in case of a breakdown situation.

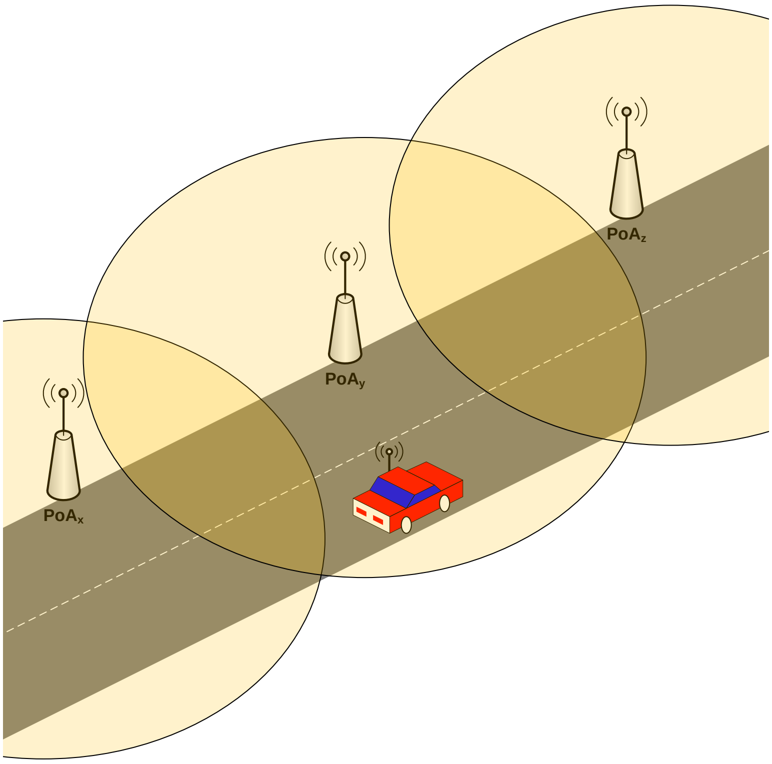

Figure 3.2: Handover scenario between WiFi networks

3.3.2 Handover between WiFi networks with a single-interface MN

As previously described, IEEE 802.21 standard facilitates handover also for single-interface MNs (i.e., hard handover). In fact, the MIES residing on the MN, if opportunely configured, can lunch a trigger when the MN is leaving the coverage area of the serving PoA. This way the MIHF can query the available MIIS server to obtain the next available WiFi network through the associated interface before the link goes down.

Once such information has been obtained, the WiFi interface can be configured to switch to the next available PoA. Thus link interruption is minimized, avoiding the need of a wireless scan. However, there is a brief time where the link is down and the MN is not associated with any PoAs, but nothing can be done to resolve this issue.

3.3.3 Handover between WiFi and 3G networks

This scenario, illustrated in Figure 3.3, describes an urban environment where users have Internet access either by means of WiFi WLAN at home, WiFi hotspots in hotels or 3G networks. WLANs are usually preferred because they guarantee a higher data rate at a lower cost. The problem is that, due to small service area, the 3G network can be a better choice for users that have high mobility.

This architecture can, for example, enable the support to IPTV, VoIP communication, email and web browsing and remote healthcare monitoring all over the city.

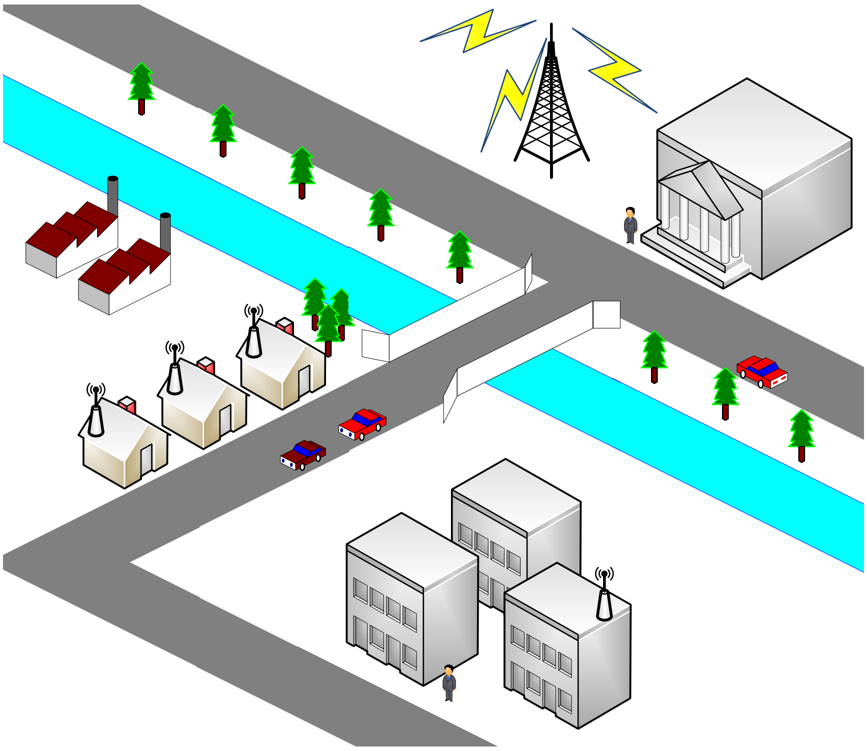

Figure 3.3: Handover scenario between WiFi and 3G networks

3.3.4 Handover between WiFi and WiMAX networks

Although the IEEE 802.11 and IEEE 802.16 WGs have defined standards to support seamless mobility within their own networks, neither WG addresses handover across these two technologies. The IEEE 802.21 WG is filling this gap by providing a media independent framework to support smooth handover. Moreover, the IEEE 802.11 TG “u” and the IEEE 802.16 TG “g” are working to enable the integration of these standards with the IEEE 802.21 framework.

In the past, several solutions have been proposed to enable this integration. In [11] an architecture is described where 802.16 Base Stations (BSs) periodically broadcast the information on the density of WiFi APs within their cell coverage. By using this information, a MN can decide the scan interval of the WLANs. This specific solution needs to modify the message format and, hence, may cause compatibility problems with existing devices.

An example scenario where WiFi and WiMAX networks may coexist is a campus university, which provides near-ubiquitous wireless coverage. Students who live on campus want to be able to move around and have their network access follow them. They have access to Internet and to the online course material, through the WLANs present in classrooms, libraries and cafes. When they decide to go out, the MN automatically switches the link connection before the source link is broken. Outside connection is enabled by the WiMAX PoAs, which create overlapping coverage areas all over the campus. Thus the students are free to move around, having instant access to email, Internet and other services, irrespective of their location.

The Northern Michigan University (NMU)[2] has deployed a WiMAX network[3] so as to offer Internet connection to university students and employees as well as some governmental entities. In fact, the school is in a rural area and commuter students and staff may live several miles away from the campus. Moreover, the university is also providing some brand new WiMAX-equipped notebooks to students, and it is also making a range of laptop and desktop WiMAX adapters available to students with non WiMAX-enabled computers. However, even though the WiMAX network is visible on campus and usable, it is not meant for wireless access when on campus, but it should be used only when WiFi is not available. Intel and NMU are currently working on an application that will automatically select the correct wireless network when the laptop is on.

---

[2] http://www.nmu.edu/.

[3] http://webb.nmu.edu/SiteSections/WiMAX.shtml.

Chapter 4

Proposed solution

As previously introduced, the objective of this thesis is the design and implementation of a solution to enable network-assisted handover. This chapter gives an in depth description of the system developed.

4.1 MIIS server

The MIIS server provides a framework for MIH entities to discover information useful for making handover decisions. It contains specific information about networks within a geographical area to enable more effective handover decision making and execution. Depending on the size of the geographical area in question and the spread of the PoAs there, the MIIS server can be distributed on several network entities. This solution is better than a centralized architecture for several reasons:

- It increases robustness because it removes the single point of failure of the system.
- It garantuees a faster response beacuse the user's requests are distributed on all the MIIS servers.

In a distributed architecture, each MIIS contains only the information related to its service area, and, optionally, redundant information for that PoAs, which are, for instance, near the service area border of two neighboring MIIS servers. One issue might be the update of the redundant information. However, this is not analyzed here because it is out of the scope of this work.

The MIIS server has been implemented as an MIH user and it has a local database which contains network information essential for handover decisions. In view of its flexibility and strength RDF was used to describe and represent network information. Thus, the MIIS supports SPARQL query language to retrieve information contained in the local database, assuring an efficient response even to complex queries.

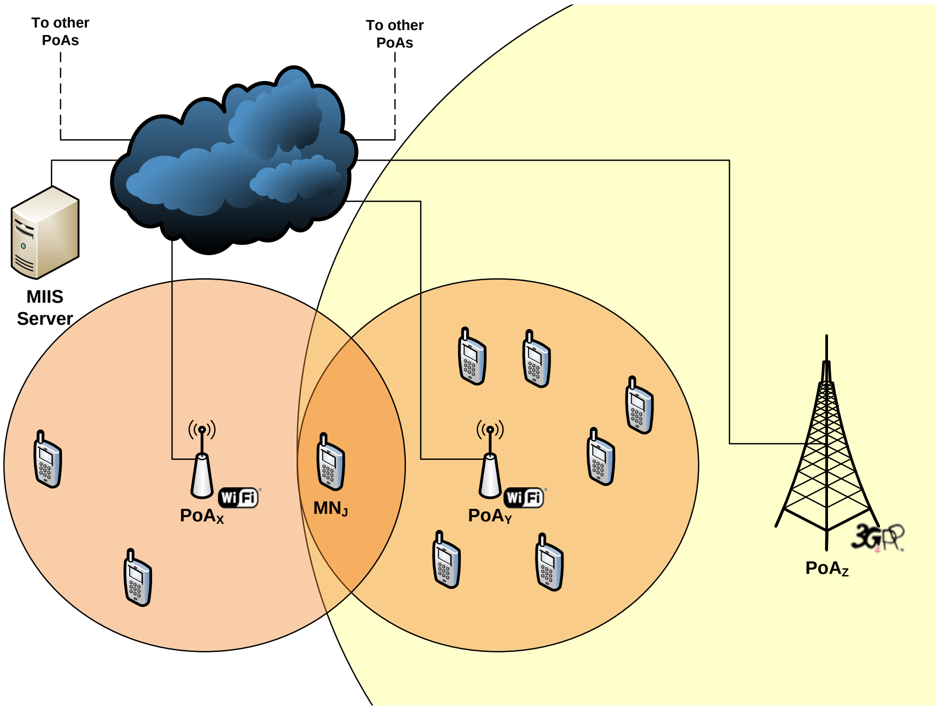

Figure 4.1: Example scenario with heterogenous overlapping wireless networks

Figure 4.1 shows an example scenario with three overlapping heterogeneous wireless networs that will be used to describe how the MIIS server takes handover decisions.

The MIIS server has three data structures, that it uses to take network-wide handover decisions. First, in the *connectivity graph* each PoA is represented as a node and the weight of the edge between node i and node j is proportional to the likelihood that an MN currently associated to node i will roam to node j. For instance, the network operator can initialize the graph based on the location of PoAs and their technology, then learning algorithms can be implemented in the MIIS server to adapt the weights based on the actual network dynamics (e.g., if handover of MNs between two nodes repeatedly fails, then the corresponding weight can be reduced or the edge in the graph can be removed).

Figure 4.2 shows the connectivity graph of the scenario shown in Figure 4.1. This graph implies that WiFi APs are preferred to the 3G PoA. It is important that the connectivity graph is updated regularly because, otherwise, the handover decision might not be up-to-date. The fact is that the exactness of this graph will affect the accuracy of the response that the MIIS server will send to a PoA when it seeks candidate PoAs. Details on how the graph is initialized and updated are outside the scope of this work.

On the other hand, the *utilization table* stores for each PoA a value between 0 and 1, where 0 means totally unused and 1 means overloaded. Table 4.1 shows

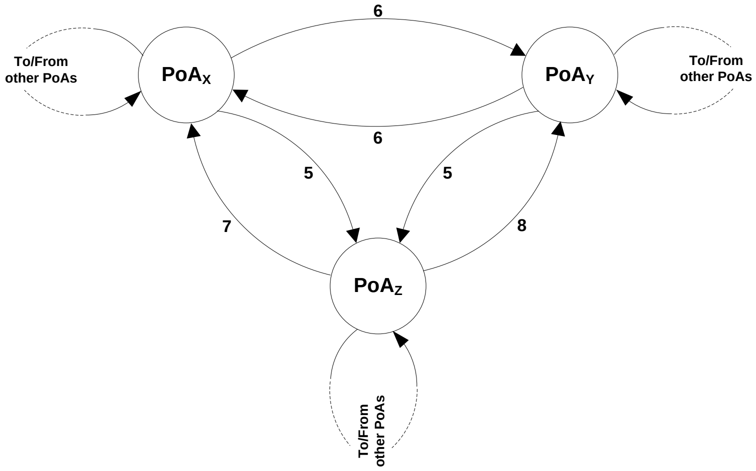

Figure 4.2: Connectivity graph of the example scenario

the utilization table of the scenario shown in Figure 4.1. We can see that PoA_z is totally unused because all the MNs are associated either to PoA_x or PoA_y.

Moreover, assume that PoA_y is overloaded. The way this value is computed depends on the link-layer technology; the table is kept up-to-date by the PoAs themselves, which piggyback the current utilization value in the MIH_Get_Information.request messages (see detailed procedure in 4.3.4). The scope of this table is to load balance the MNs among available PoAs in order to get optimal resource utilization, maximize throughput, minimize response time and avoid overload. This fact, in conjunction with the distribution of the MIIS server on distributed network entities, should avoid overloaded situations on the network. Thus, a non overloaded PoA might be selected as the best PoA for an MN at a particular time, also if the weight on the connectivity graph is less than an overloaded one.

By combining the data in the connectivity graph and the utilization table, the MIIS server can take handover decisions for the MNs taking into account both long-term and short-term inputs, respectively. For instance, as shown in Figure 4.1, assume that the MN_j is associated with PoA_x and that it is moving towards PoA_y. If the MIIS server uses only the connectivity graph for the handover decision, it suggests PoA_y to the MN_j, as the next target PoA. The problem is that this AP is overloaded, as can be seen from the Table 4.1. This way, combining the data in the connectivity graph and the utilization table, the MIIS server selects PoA_z as the best PoA for the MN_j.

One of the main problem of the MIIS server is that it knows only the area where an MN associated to a particular PoA currently resides. However, the MN is generally free to move around and it does not necessarily follow a precise path.

Table 4.1: Utilization table of the example scenario

PoA ID	**Utilization Value**
PoA_x	0.3
PoA_y	0.6
PoA_z	0
...	...

Thus the problem is to supply the associated PoA with the list of candidate PoAs for that MN when the handover becomes necessary. The MIIS server uses the connectivity graph to try to overcome this problem, but this is not sufficient to guarantee that the suggested PoAs will be available to the MN when it performs handover. In fact, the graph depends only on the disposition and the technology of the PoAs in the network. The weight of the edge between two nodes is based on several factors as previously described, but it does not depend on the MN which is currently involved in handover. This way, the MIIS should be able to predict the movement of the MN in order effectively to find the next best PoA. In [24] the authors propose three WLAN discovery schemes which are based on the IEEE 802.21 standard so as to minimize energy consumption.

More accurate predictions might be done using the information gathered from the Global Positioning System (GPS) interface of the MN. However, in some environments where MNs follow a specific direction and do not roam in a random way (e.g., a motorway, where the vehicles go along the road and can not turn back), the MIIS server can more easily predict the next PoA. The serving PoA and the direction of the MN is the only information that it needs. The MN direction can easily be calculated taking into account the latest PoAs that the MN was associated to. The serving PoA is indicated by the PoA itself, when it queries the MIIS server.

Figure 3.2 above shows an example of a motorway with three overlapping PoAs. Assume that the MN is associated with PoA_y and that it is going from left to the right. Moreover, the weight between PoA_y and PoA_x is higher than the weight between PoA_y and PoA_z. Thus, if the MIIS server uses only the connectivity graph for a handover decision, it suggests PoA_x to the MN, as the next target PoA. However, the MN was previously associated with PoA_x, so the MN should choose PoA_z, as next target PoA. This situation does not occur if the MIIS server remembers the last PoA for each MN, because it puts this PoA as the last choice for the MN. For this reason, the MIIS server has another data structure, where it stores the last PoA for each MN residing in its service area.

The MIIS server must first register to the local MIHF to receive the messages sent by the remote PoSs. When it receives a MIH_Get_Information.request, it queries its local database and generates a XML response that is sent by the MIHF to the originator of the request. However, to use the services supplied by the MIIS, a remote entity must first register itself with the MIHF of the

MIIS. After a successful registration, the MIIS sends the list of supported RDF schema Uniform Resource Locators (URLs) to the just registered MIH user, with a MIH_Push_Information.request, in order to inform the other entities about supported data. After this step, the MIH user can query the MIIS to obtain useful information about candidate PoAs for an MN or neighboring network maps.

4.1.1 SPARQL/Update

SPARQL/Update[1] is a language to express updates to an RDF store. It uses a syntax derived from SPARQL and is envisaged to be used in conjunction with the SPARQL query language. SPARQL/Update provides the following facilities:

- Insert new triples to an RDF graph.
- Delete triples from an RDF graph.
- Create a new RDF Graph to a Graph Store.
- Delete an RDF graph from a Graph Store.

A SPARQL/Update query is used by the serving PoA to keep its value in the utilization table of the MIIS server up-to-date. This query is piggybacked in the MIH_Get_Information.request, when the PoA queries the MIIS to obtain the candidate PoAs for an MN.

4.1.2 Vendor-specific information elements

In order to enable the proposed network-assisted handover solution, the RDF basic schema defined by the IEEE 802.21 standard is no longer sufficient and it must be extended. This way, a RDF extended schema has been defined, containing vendor-specific IEs and containers of IEs. The IEs defined in the standard enable a network operator to describe the list of PoAs for each supported network and for each PoA there are IEs which describe specifc PoA properties, such as link-layer addresses and geographical location. Multiple location types are supported, including coordinate-based location information (e.g., XML-formatted geospatial location) and cell ID, to uniquely identify a cell within a 3GPP UMTS Terrestrial Radio Access Network (UTRAN). The vendor-specific IEs containers defined are the following:

- *IE_CONTAINER_LIST_OF_NEIGHBOR_POAS*, which contains the list of neighboring PoAs (i.e., list of IE_CONTAINER_NEIGHBOR_POA) for each PoA in the network. This information is used by the MIIS server to select the candidate PoAs in the area where the MN currently resides.

---

[1] http://www.w3.org/Submission/SPARQL-Update/.

- *IE_CONTAINER_NEIGHBOR_POA*, which contains the information that depicts a neighboring PoA (i.e., link address of the PoA and a weight value used to create the connectivity graph). Towards this information, the MIIS server can uniquely identify a PoA in order to retrieve from the local database other useful information.

The complete definition of the RDF extended schema is in Appendix A. Table 4.2 shows the vendor-specific IEs defined with their description.

Table 4.2: Vendor-specific Information Elements

Information Element	**Description**
IE_POAD_LOAD	It represents the current load on a PoA. This information is piggybacked in the MIH_Get_Information.request message to keep the utilization table of the MIIS server up-to-date.
IE_POA_MIHF_ID	It indicates the MIHF ID of a PoA. This information is used by the MIIS server to know the serving PoA.
IE_POA_QUERIER_ID	It indicates the MIHF ID of the MN involved in handover. For instance, the MIIS server can use this information to retrieve the last PoA of the MN from the local database.
IE_LINK_ADDR	It indicates the link address of a neighboring PoA. This information is used by the MIIS server to construct the connectivity graph.
IE_WEIGHT	It indicates the weight of the edge between two PoAs. This information is used by the MIIS server to construct the connectivity graph.

4.2 Overview of network-assisted handover procedures

In this section, the proposed network-assisted handover procedure will be described and more detailes will be given in Section 4.3.

The logical view of a sample network with two PoAs, with co-located PoSs, and the MIIS server is illustrated in Figure 4.3, which also shows the MIHF, the MIH_USR (i.e., the MIH user), and MIH_LINK_SAP entities in all the network elements involved. Note that all the MIHF blocks of the infrastructure nodes, including the MIIS node, are bound together in a full mesh mode. This is to allow network-to-network exchanges to favor the handover, and it is done by means of automatic discovery procedures. How such connections are created is not relevant to this work and, hence, is not discussed here.

From a high level perspective, the procedures are as follows. When the MN first enters the network (e.g., it is switched on), it enters a *bootstrap* stage. This is the only phase when the MN is required to scan through all the interfaces to find a suitable PoA. After association and IP connectivity establishment, the MIHF of the

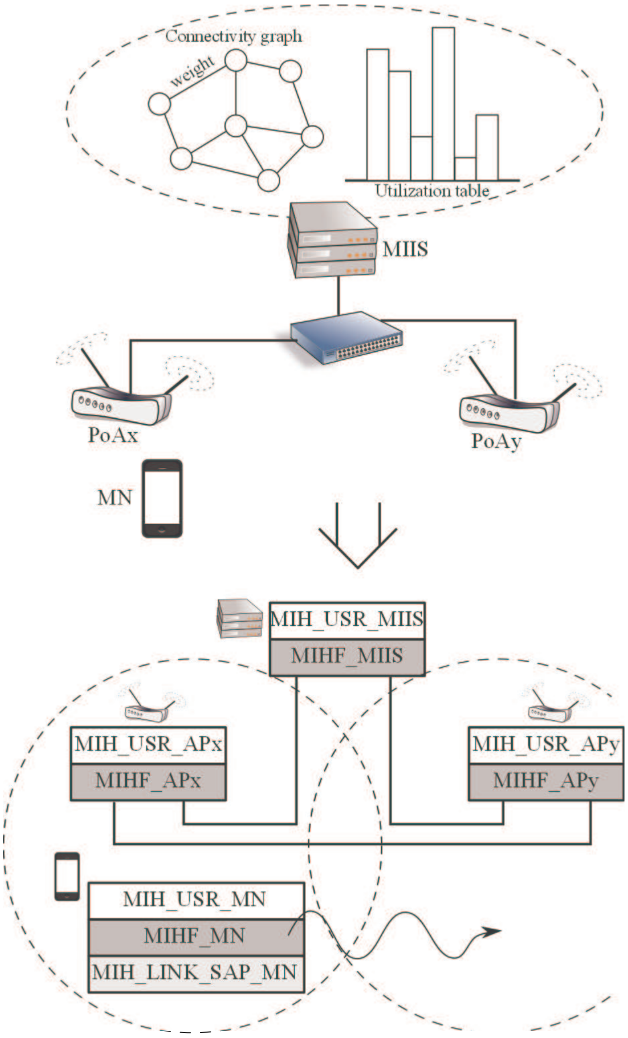

Figure 4.3: Logical view of the MIH elements in a sample network

MN discovers the MIHF of the PoA and binds it to itself. In this implementation, discovery is carried out by means of UDP broadcast. After binding, the MIHF of the PoA registers to link parameter events, so as to be kept informed on changing conditions that might lead to a handover, and sends a command to switch off the unused wireless interfaces, so as to reduce power consumption. Until the MN remains in the network managed by the MIIS server, it will never be required to perform scan again and it will use only one network interface at a time, except for a brief moment when handover occurs.

The situation does not change until a handover occurs. The handover procedure is triggered by the MIH_LINK_SAP of the MN, which sends an event to the MIHF_MN when the link quality of the associated interface is degrading and, hence, a handover is imminent.

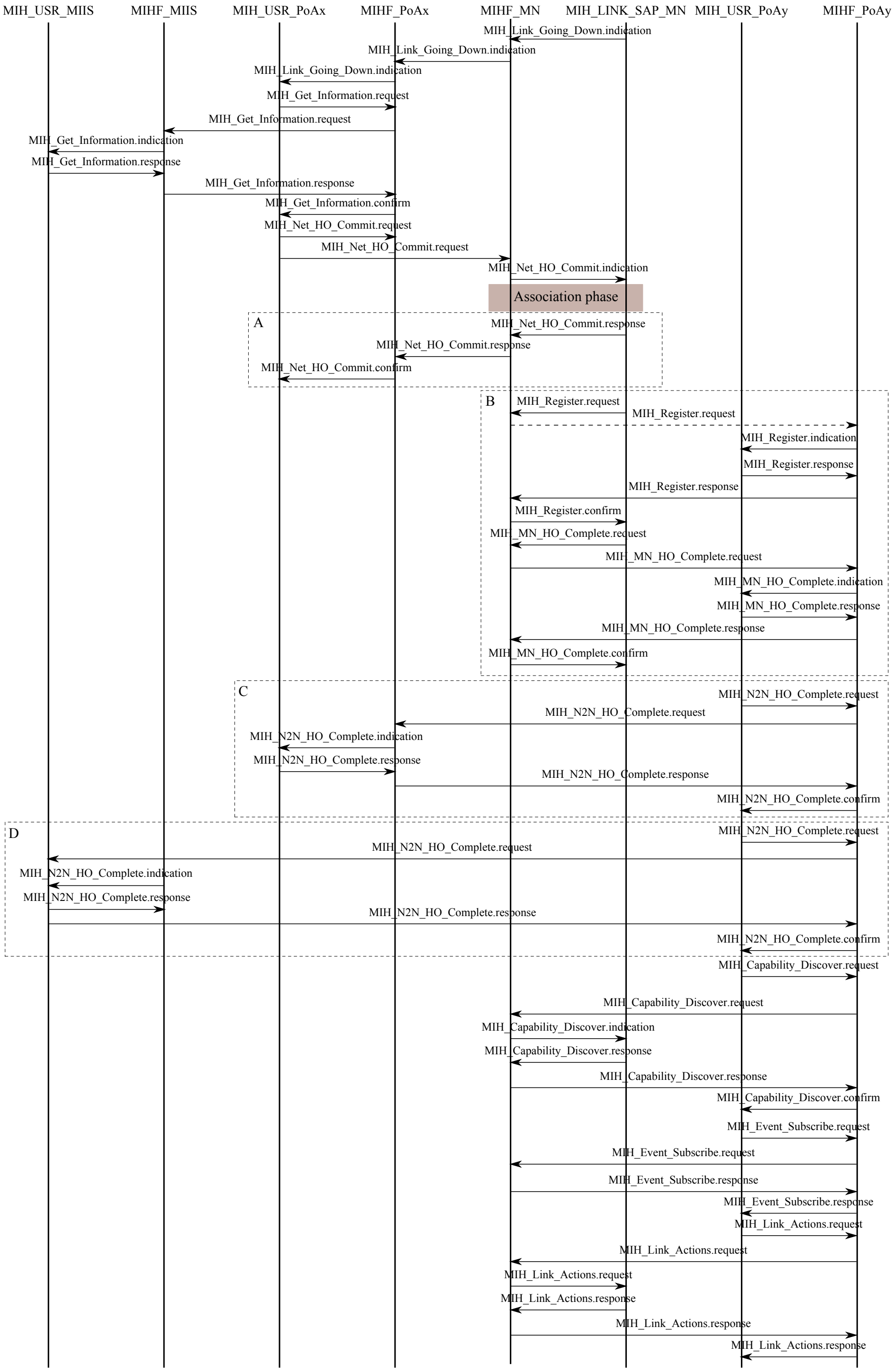

Figure 4.4: Sequence diagram of a network-assisted handover

This means that the MN is leaving the coverage area of the serving PoA. The precise definition of the link quality degradation is technology-dependent and must be implemented in the link-layer specific part of the MIH_LINK_SAP. The trigger is conveyed by the MIHF_MN to the PoA, which performs a SPARQL query on the MIIS server to retrieve the set of candidate PoAs. The request message contains also a SPARQL/Update query used by the serving PoA to keep its value in the utilization table of the MIIS server up-to-date.

The query result from the MIIS, encoded in XML[2], is based on its internal handover algorithm, which takes as input the connectivity graph, the utilization table, the last and the serving PoA of the MN. In this current implementation, a simple baseline algorithm is employed, which seeks the candidate PoAs by pruning all the PoAs with their utilization above a configurable threshold and sorts the remaining ones in decreasing order of their weights in the connectivity graph. If the last PoA of the MN is in the candidate list, it is indicated as last choice. The ordered list of PoAs is passed to the serving PoA and, eventually, to the MN. Then, the MIHF of the MN tries all the candidate PoAs until association succeeds with one of them, which becomes the target PoA.

At this point, the MN notifies both the serving PoA and the target PoA, which now becomes the serving PoA. The latter, also completes the handover procedures with network-to-network messages with both the previous PoA (in case the MN could not notify it due to excessive link quality degradation or, in the case of hard handover, if the source link is not available any more) and the MIIS server (so that it can learn from the outcome of its handover decisions).

The detailed sequence diagram is illustrated in Figure 4.4, where PoA_x is the serving PoA and PoA_y is the target. In the diagram some blocks of messages are enclosed with dashed boxes marked with letters from A to D, because they can happen simultaneously and, hence, their order in the figure is arbitrary.

4.3 Detailed network-assisted handover procedures

In the following section, every phase of the handover procedure is fully analyzed.

4.3.1 Bootstrap phase

When the MN first enters the network, it does a bootstrap. It makes a scan through the available interfaces until association succeeds. The MN is assumed to be a non smart device, so when it is switched on, a wireless scan is the only thing that it can do. Other handover decisions are network-driven. The bootstrap stage is the only phase where the MN is required to scan to find a suitable PoA. In the other phases the network itself informs the MN about availabe PoAs through specific IEEE 802.21 messages. However, if the MN loses connection before the network has provided suitable neighoring PoAs, it enters again in a bootstrap stage.

---

[2] http://www.w3.org/TR/rdf-sparql-XMLres/.

As illustrated in Figure 4.3, the MN is composed by three modules: the MIH_LINK_SAP_MN, the MIHF_MN and the MIH_USR_MN. The latter communicates with the remote network entities that reside in the various PoAs, in order to be assisted during handover. When the MN is powered on, the MIH_USR_MN binds the local MIHF to itself and gets the list of available interfaces on the MN by means of a MIH_Capability_Discover.request.

When it receives the MIH_Capability_Discover.response message, it sends a MIH_Link_Actions.request message to the MIHF_MN to begin a wireless scan on one of the network interfaces. The MIH_Capability_Discover.response contains the following fields:

- Status of the operation, such as failure or success. This is the only required field. The others are optional.
- List of network interfaces on the device. Information contained in the message, such as the link address (e.g., MAC address) and supported network types (e.g., IEEE 802.11 a,b and g) are more detailed than those contained in the MIH_Register.request message.
- List of supported MIH events on local MIHF (e.g., MIH_Link_Up and MIH_Link_Down).
- List of supported MIH commands on local MIHF (e.g., MIH_Link_Actions and MIH_Link_Configure_Thresholds).
- List of supported MIIS query types on local MIHF (e.g., SPARQL or TLV query).
- List of supported transport types on local MIHF (e.g., UDP or TCP transport).
- List of network type pair on which soft handover is supported. (e.g., from IEEE 802.11 to IEEE 802.16)

The MIH_Link_Actions.request contains the list of suggested actions to be executed on the MN. Due to the flexibility of the IEEE 802.21 standard, vendor-specific actions can be defined. Each action is composed of the following fields:

- The interface on which the action is performed.
- The specific action to be executed. It can indicate either to perform a wireless scan or to power on/off the interface.
- Time to elapse before an action needs to be taken. A value of 0 indicates that the action will be taken immediately.

Afterwards, the MIHF_MN, using the primitives provided by the MIH_LINK_SAP, sends the MIH_Link_Actions.response message to the MIH_USR_MN with the result of the wireless scan.

The MIH_Link_Actions.response contains the result of the requested actions, each of which has the following fields:

- The interface on which the action has been executed.
- The result of the action, such as success, failure or refused.
- If the request message specifies a wireless scan, this field contains the scan result. For each found PoA, the link address (e.g., the MAC address), the network ID (e.g., the SSID of a WLAN) and the signal strength are indicated.

Finally, the MIH user of the MN, through the MIH_Link_Actions.request, tries all PoAs in the list until association succeeds with one of them. If the procedure fails for all the candidate PoAs, this process is repeated on another interface. After association and IP connectivity establishment, the bootstrap phase ends.

4.3.2 MN registration

After association and IP connectivity establishment, the MIHF of the MN has to declare its presence to the MIHF of the PoA. This is done by broadcasting a MIH_Register.request because the MIH_USR_MN does not know the IP address and the MIHF ID of the serving PoS. UDP port, used for MIHF communication, is the only information known by the MN.

Once the message is received by the remote MIHF_AP, it registers the MIHF_MN in its data structure, so as to enable future MIHF communication. The MIH_Register.request is then passed to the MIH_USR_AP as an indication message. This way, the MIH user of the serving PoA can therefore discover supported services on the MN. The MIH_Register.request contains the following information:

- Source MIHF ID, which is used by the receiver MIHF to register the peer, so as to enable future communication with the remote MIHF.
- List of available interfaces on the remote MIHF.
- Registration request code, which can either indicate a registration or a re-registration if, for instance, the previous registration has expired.

Finally, the MIH_USR_AP sends the processing status of the received registration request to the remote MIHF, by means a MIH_Register.response, indicating the validity period of the registration. The registration process is complete when the MIHF_MN receives the response message, which is passed to the MIH_USR_MN as a confirmation message. A MIH_DeRegister.request can be later used by a MIH user to de-register the local MIHF from an MIHF peer.

4.3.3 Capability discover and event subscription

MIHF registration is the first step necessary to inform the serving PoS about the presence of the MN, so as to be later supported on handover decisions. After a successful registration, the MIH_USR of the serving PoA sends a MIH_Capability_Discover.request message to the MIHF_MN to discover the supported MIH services and the available network interfaces. The capability information of the local MIHF can also be piggybacked in this request, so that the two MIHF entities can mutually discover each other's capabilities with a single invocation of this primitive. However, in this context, this information is not useful to the MN, so it is not present in the MIH_Capability_Discover.request. When the MN receives the message, it responds with a MIH_Capability_Discover.response to convey the locally supported MIH capabilities and the available interfaces to the serving MIH_USR_AP.

Afterwards, the serving PoA registers to link parameter events on the associated interface, so as to be kept informed on changing conditions that might lead to an handover, by means of a MIH_Event_Subscribe.request. This message contains the remote interface interested in event subscription and a list of MIH events that the serving PoA would like to receive indications for, such as MIH_Link_Going_Down and MIH_Link_Paramters_Report events. Optionally, the subscriber can indicate a list of specific configuration information applicable for various events being subscribed.

For example, the MIH_USR_AP could set a threshold on the Received Signal Strength Indication (RSSI) of the associated interface, so as to receive a MIH_Link_Parameters_Report.indication when the given threshold is crossed. When the serving PoA is no longer interested in receiving those notifications, it can unsubscribe by means of a MIH_Event_Unsubscribe.request. The event subscription is complete when the requesting MIHF_AP receives a MIH_Event_Subscribe.response from the MIHF of the MN.

Another way to set parameter report thresholds on the MN is towards a MIH_Link_Configure_Thresholds.request. For instance, the MIH_USR of the serving PoA could set a threshold on the signal level of the associated interface of the MN. When the threshold is crossed, the notification is received by that PoA. This can mean that the signal is weak, and, hence, that a candidate network may be able to provide a better QoE for the user. The serving PoA may not configure any threshold, if the MIHF of the MN supports the MIH_Link_Going_Down event. In fact, when it receives a MIH_Link_Going_Down.indication, it means that the MN is leaving its coverage area and handover is required. The main difference between the two methods is that, with the first, the serving PoA can specify a well defined signal level which can not be crossed if it is to guarantee a particular QoS; the MIH_Link_Going_Down.indication is instead generated by the MN and it is implementation-dependent.

A MIH_Link_Parameters_Report is also employed for issuing periodical notifications about link conditions, for example, to constantly control the QoS on a particular link. Link parameter events can be used by an MIH user to initiate the handover candidate discovery process, or trigger applications to adapt to changing link conditions.

After subscription, the serving MIH_USR_AP sends a command to switch off the unused interfaces on the MN, by means of a MIH_Link_Actions.request, so as to reduce power consumption.

4.3.4 Handover procedures

The handover procedure is triggered by the MIH_LINK_SAP of the MN, which sends a notification to the MIHF_MN when the link quality is degrading and, hence, a handover is imminent. Afterwards, the MIHF_MN sends a MIH_Link_Going_Down.indication to the subscribed remote MIHF (i.e., the MIHF_AP of the serving PoA). This notification contains the identifier of the interface associated with the event, the time interval at which the link is expected to go down and the reason why the link is going down (e.g., a link parameter degrading that may soon lead to the breaking of the link).

When the message is received by the MIH_USR_AP of the serving PoA, it checks whether the interface belongs to an associated MN. If so, it prepares a MIH_Get_Information.request containing a set of SPARQL queries to discover neighboring PoAs, which might subsequently serve the MN. The query used to retrieve the set of candidates PoAs contains the following information:

- MIHF ID of the MN involved in handover. This information is used by the MIIS server to retrieve the last PoA of the MN from the internal table. Moreover, the network operator could recommend different PoAs depending on the contract chosen by the user. For example, a gold user may be directed to a faster and less loaded PoA than the one chosen for a silver user.

- The list of network interfaces available on the MN (e.g., IEEE 802.11g or IEEE 802.16 interface). This information is used by the MIIS server to retrieve, from the list of neighboring PoAs of the serving PoA, the only ones which are compatible with the technologies supported by the MN. For instance, if the MN has two interfaces, an IEEE 802.11g and IEEE 802.16 interface, the MIIS server will select only those PoAs which support those interfaces.

- The serving PoA, which is used by the MIIS to obtain the list of candidate PoAs.

The MIH_Get_Information.request contains also a SPARQL/Update query, which is used by the serving PoA to keep its value in the utilization table of the MIIS server up-to-date. In fact, in order to limit the number of messages sent to the server, the PoA piggybacks its current load on other requests. Then, the message is sent to the MIHF of the MIIS server, which passes it to the MIH_USR_MIIS as an indication message.

The query result from the MIIS server, encoded in XML, is based on its internal handover algorithm, which takes as input the connectivity graph, the utilization table and the serving and last PoA of the MN. In this current implementation, a simple baseline algorithm is employed, which seeks the candidate PoAs by pruning all the PoAs with their utilization above a configurable threshold and sorts the remaining ones in decreasing order of their weights in the connectivity graph. The query result is then inserted in the MIH_Get_Information.response and is sent by the MIH_USR_MIIS to the MIH_USR_AP of the serving PoA, towards the local MIHF.

Afterwards, the serving PoA sends the ordered list of candidate PoAs to the MN, by means of a MIH_Net_HO_Commit.request. This primitive is employed by the serving PoA to request from the MN the commitment to perform a network-assisted handover based on selected choices for PoAs. This message is then passed to the MIH_USR_MN as an indication message and it contains the following fields:

- The link type of candidate PoAs.
- The ordered list of PoAs for assisting the MN to perform the handover. This list requires vendor-specific extensions, which can easily be accommodated due to the flexibility of IEEE 802.21 messages. In fact, this field contains the network identifier, the link address and optionally a type to represent an auxiliary access network identifier (e.g. Homogeneous Extended Service Set ID (HESSID) if network type is IEEE 802.11) for each PoA in the list. In order to configure the interface of the MN properly before handover begins, other information is needed, such as the frequency of an IEEE 802.11 AP.
- The set of resource parameters assigned to the MN for performing the handover. This field includes both QoS parameters reserved to the MN on the target PoA and the required configuration of the reserved resources at the target network. In this impementation this field is not used.
- Optionally, a list of network controlled handover actions for the link and the time to elapse before those actions need to be taken.

Finally, the MIHF of the MN tries all the candidate PoAs in the ordered list until association succeeds with one of them, which becomes the target PoA. At this point, the MN notifies both the serving PoA and the target PoA, which will now become the serving PoA, using a MIH_Net_HO_Commit.response and a MIH_MN_HO_Complete.request message respectively. The latter also completes the handover procedures with network-to-network messages to and from both the previous PoA using a MIH_Net_HO_Commit.response message (in case the MN could not notify it due to excessive link quality degradation) and the MIIS server (so that it can learn from the outcome of its handover decisions) by means of a MIH_N2N_HO_Commit.request.

The MIH_N2N_HO_Complete.request message is used by the MIH user of a PoA to communicate with a peer network MIH entity (e.g., MIH_USR_AP or MIH_USR_MIIS) about the completion of handover operations and it contains the following fields:

- The MIHF ID of the MN, to indicate the MN that has completed the handover procedures.
- The source and the target link of the handover operation. This information respectively indicates the source and the target PoA involved in handover.
- The handover procedure result.

The MIH_MN_HO_Complete.request message is used by the MIH user of an MN to communicate with a peer network MIH entity (e.g., MIH_USR_AP) about the completion of handover operation and it contains the same fields as the MIH_N2N_HO_Complete.request message, except for the MIHF ID of the MN, which is automatically obtained from the message.

4.3.4.1 Example of SPARQL/Update query

SPARQL/Update query is used by an MIH user of a PoA to keep its value in the utilization table of the MIIS server up-to-date. Listing 4.1 shows an example of a SPARQL/Update query contained in the MIH_Get_Information.request message. The PoA, identified by *MIHF_AP_ID*, updates its current load to the value specified by *CURRENT_LOAD*.

Listing 4.1: SPARQL/Update query example

```
PREFIX mihbasic: <BASE_SCHEMA_URL#>
PREFIX mihext: <EXTENDED_SCHEMA_URL#>
MODIFY
DELETE
{ ?container_poa mihext:ie_poa_load ?load . }
INSERT
{ ?container_poa mihext:ie_poa_load "CURRENT_LOAD" . }
WHERE
{ "MIHF_AP_ID" mihbasic:ie_container_poa ?container_poa
    . }
```

4.3.4.2 Example of SPARQL query request

Listing 4.2 shows an example of a SPARQL query which is contained in the MIH_Get_Information.request message. This query is used by the MIH_USR of the serving PoA to obtain the set of candidate PoAs for an MN, whose MIHF is identified by the *MIHF_MN_ID*. In this example, it is assumed that the MN supports IEEE 802.11 a,b and g networks. This way, the serving PoA, identified by the *AP_MAC_ADDRESS*, requests to the MIIS server only the list of candidate PoAs which supports these technologies. The query result, encoded in XML, is put by the MIIS server in the MIH_Get_Information.response message and it is sent to the serving PoA.

Listing 4.2: SPARQL query example

```
PREFIX mihbasic: <BASE_SCHEMA_URL#>
PREFIX mihext: <EXTENDED_SCHEMA_URL#>
SELECT ?n_id, ?poa_l_addr, ?l_ch_range, ?h_ch_range
WHERE {
?network mihbasic:ie_network_id ?network_id .
?network mihbasic:ie_network_type ?net_type .
?net_type mihbasic:link_type "19" .
?net_type mihbasic:subtype ?s_type .
?network mihbasic:ie_container_poa ?container_poa .
?container_poa mihext:ie_poa_querier_id "MIHF_MN_ID" .
?container_poa mihbasic:ie_poa_link_addr  ?link_addr .
?link_addr mihbasic:mac_addr "AP_MAC_ADDRESS" .
?container_poa mihext:ie_container_list_of_neighbor_poas
     ?neighbor_list .
?neighbor_list mihext:ie_container_neighbor_poa ?
   neighbor_poa .
?neighbor_poa  mihext:ie_link_addr ?poa_l_addr .
?l_addr mihbasic:mac_addr ?poa_l_addr .
?new_container_poa mihbasic:ie_poa_link_addr ?l_addr .
?new_container_poa mihbasic:ie_poa_channel_range ?
   channel_range .
?channel_range mihbasic:low_ch_range ?l_ch_range .
?channel_range mihbasic:high_ch_range ?h_ch_range .
?n mihbasic:ie_container_poa ?new_container_poa .
?n mihbasic:ie_network_id ?n_id .
FILTER (?s_type <= "1") .
}
```

Chapter 5

Performance evaluation

5.1 Implementation

The proposed network-assisted handover solution has been implemented in a small-scale testbed. The MIHF developed within the ODTONE[1] project was employed, with some necessary modifications for optimization and fine tuning, and for extending those functional modules needed but not fully implemented (e.g., MIHF registration and discovery). The MIH users of the PoAs and the MIIS server, as well as the MIH_LINK_SAP and the MIH user of the MNs, were also implemented using the MIHF and the Application Programming Interfaces (APIs) provided by the project.

5.1.1 ODTONE project

Figure 5.1: ODTONE project logo

ODTONE stands for Open Dot Twenty ONE and is an open source implementation of the Media Independent Handover framework from the IEEE 802.21 Media Independent Handover Services standard, using C++ APIs. The project started in July 2009 at the University of Aveiro, in Portugal, and it is released under the Lesser General Public License (LGPL) license[2]. ODTONE aims to implement an MIHF that is capable of being deployed in multiple operating systems, such as GNU/Linux, Microsoft Windows and Android. ODTONE supplies the implementation of an MIHF, supporting its inherent services (MIES, MIIS and MICS, as

---

[1] http://helios.av.it.pt/projects/odtone/.

[2] http://www.gnu.org/licenses/lgpl.html.

well as supporting mechanisms (e.g., event registration). ODTONE's implementation provides an MIHF which works as a base for the user's scenarios and enables the users to implement their own MIH users, MIH_SAP and MIH_LINK_SAP. ODTONE provides a simple and flexible interface for the development of these SAPs, handling MIH Protocol messages and state transitions. Note that Boost C++ libraries[3] (version 1.37 or higher) are required to compile ODTONE.

A modified version of the current release of the project (i.e., ODTONE 0.2, issued April 26, 2010) was used to implement the network-assisted handover solution described in Chapter 4.

5.1.2 MIIS server architecture

The MIIS server is composed of two modules, as shown in Figure 4.3, i.e., the MIHF_MIIS and the MIH_USR_MIIS. The MIH_LINK_SAP is not necessary since the server does not need to interact with the layers below the MIHF, for instance, to configure a network interface or to receive notifications from the network devices. The main function of the MIHF_MIIS is to pass all the messages received from MIHF peers (e.g., MIHF_AP) to the MIIS server and to handle MIHF registration, in order to enable communication among remote MIH users. The MIIS server was implemented as an MIH user (i.e., MIH_USR_MIIS) and it has a local database which contains network information essential for handover decisions. It uses the framework supplied by the MIHF and the APIs provided by the MIH_SAP, to assist the MNs on handover decisions.

The MIIS server was implemented as a multi-thread process, as shown in Figure 5.2, so as to handle multiple requests from the PoAs in parallel. One thread waits for incoming messages from the local MIHF; whenever a new message arrives, it creates a new thread to process the received message. The following messages are supported by the MIIS server:

- **MIH_Register** A remote PoS, which wants to use the services supplied by the MIIS server, must first register with the server. This way, an MIH user on a PoA, when it is switched on, sends a registration request to the MIHF_MIIS, by means of a MIH_Register.request. This message is then passed to the MIH_USR_MIIS as an indication message and it is received by the listening thread. Once the registration process is complete, the MIH_USR_MIIS sends the response message to the registration request and the MIH_Push_Information.request to the MIH user of the remote peer. The latter contains the supported RDF schemas, which will be used by the MIH_USR_AP to discover the supported IEs.

- **MIH_DeRegister** A remote PoS, which is either no longer interested in receiving the services supplied by the server or is preparing to shut down, should first inform the MIIS server by means a MIH_DeRegister.request.

---

[3] http://www.boost.org/.

This way, when the MIHF_MIIS receives this message from a remote MIHF, it deletes this peer from the registered peers and informs the MIIS server with an indication message. Thus, the server could delete the information about this PoA, contained in the neighboring map, so as to be up-to-date. Finally, the MIH_USR_MIIS sends the MIH_DeRegister.response to the remote MIH user to complete the de-registration process.

- **MIH_N2N_HO_Complete** When a handover process is complete, the MN informs both the serving and the target PoA, which now becomes the serving PoA. The latter, also completes the handover procedures with network-to-network messages (i.e., the MIH_N2N_HO_Complete messages) to and from the MIIS server, so that it can learn from the outcome of its handover decisions. After registering the last PoA of the MN in its data structure, the MIH_USR_MIIS sends the MIH_N2N_HO_Complete.response to the MIH user of the serving PoA.

- **MIH_Get_Information** The MIH_Get_Information.request is used by the remote MIH users of the PoAs to query the MIIS server about neighboring PoA information and supported RDF schemas. For each message received, the listening thread creates a new thread which processes the request, extracts the SPARQL queries, parsing and executing them, takes the handover decisions, and prepares the XML query result to be sent in the MIH_Get_Information.response message to the requesting MIH user. The connectivity graph is loaded at startup via an XML/RDF configuration file, using a customized version of the IEEE 802.21 schema (i.e., the basic and the extended schema), and it is not modified at run time. With few modifications at the current implementation, it could be possible to dynamically update the connectivity graph, for instance, to support new PoAs attached to the network or modify the weight between two particular nodes depending on the outcome of the previous handover decisions. The utilization table is instead initialized with empty load for all PoAs and then updated according to the indications from the PoAs themselves, by means of a SPARQL/Update query contained in the MIH_Get_Information.request messages. Each MIH_Get_Information.request can contain either a list of SPARQL queries or a request for a RDF schema URL. Other queries are not supported in this implementation, i.e., TLV queries. The Multipurpose Internet Mail Extensions (MIME) field is used to differentiate the SPARQL query type (i.e., *"application/sparql-query"* or *"application/sparql-update-query"*) contained in the MIH_Get_Information.request for indicating a standard SPARQL and a SPARQL/Update query respectively. The new thread processes all the queries contained in the message, following the ordered list, and it respectively does the following actions depending on the query type:

  - If the query is a SPARQL/Update query, it updates the value in the utilization table for the specified PoA.

  - If the query is a standard SPARQL query, it executes the query. The query result, encoded in XML, is based on a handover algorithm, which takes as input the connectivity graph, the utilization table and the serving and the last PoA of the MN. In this current implementation, a baseline simple algorithm is employed, which seeks the candidate PoAs with their utilization above a configurable threshold and sorts the remaining ones in decreasing order of their weights in the connectivity graph. If the last PoA of the MN is in the ordered list of candidate PoAs, it is put as the last choice.

  The MIH_USR_MIIS uses Redland RDF libraries[4], which provide support for Resource Description Framework, e.g., for parsing and serializing RDF syntaxes and for executing RDF queries. All Redland packages are free software/open source software and are released under the LGPL 2.1[5], General Public License (GPL) 2[6] or Apache 2[7] licenses as alternatives.

The MIIS server is technology-independent and it has been developed and tested on Linux. Figure 5.2 shows the functional scheme of the MIH_USR_MIIS. The connectivity graph, the utilization table and the RDF schemas are contained in the *RDF/XML Data*. Information on the last PoA for each MN is instead contained in the data structure of the *Last PoAs*. The listening thread is composed of the following modules:

- *MIHF Registration Handler*, which deals with registration and de-registration of new MIHF entities. It is responsible for sending the MIH_Push_Information.request to the registered peers, which are informed about supported IEs.
- *Handover Notification Handler*, which manages notification on handover completion. This module is responsible for registering the last serving PoA in the internal data structure.
- *Handover Assistant*, which helps PoA in handover decisions. For each MIH_Get_Information.request received, this module creates a new thread, which processes the message. This thread is in turn composed of the following two modules:
  - *Information Updater*, which is responsible for updating the utilization table using information contained in the SPARQL/Update query.
  - *Handover Decisor*, which executes the SPARQL queries, so as to obtain the list of candidate PoAs for a particular MN.

---

[4] http://librdf.org/.

[5] http://www.gnu.org/licenses/lgpl-2.1.html.

[6] http://www.gnu.org/licenses/gpl-2.0.html.

[7] http://www.apache.org/licenses/LICENSE-2.0.html.

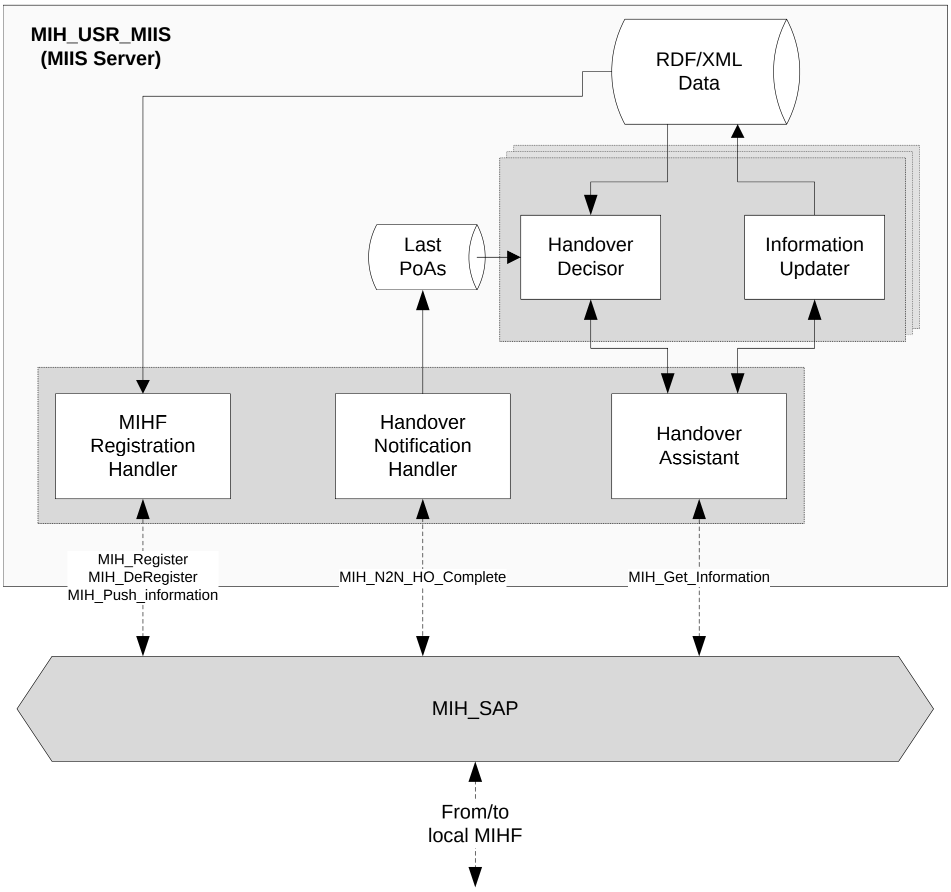

Figure 5.2: MIIS Server functional scheme

5.1.3 PoA architecture

Each PoA is composed of two modules, as shown in Figure 4.3, i.e., the MIHF_AP and the MIH_USR_AP. The MIH_LINK_SAP is not necessary since the MIH user does not need to interact with the layers below the MIHF, for instance, to configure a network interface or to receive notifications from the network devices. The main function of the local MIHF is to pass all the messages received from MIHF peers (e.g., others MIHF_AP or MIHF_MIIS) to the MIH user and to handle MIHF registration, so as to enable communication among remote MIH users.

The MIH_USR of the PoAs was implemented as a multi-thread process, as shown in Figure 5.3, so as to manage multiple requests from MNs in parallel. It uses the framework supplied by the MIHF and the APIs provided by the MIH_SAP, to assist associated MNs in handover decisions. It is technology-independent and it has been developed and tested on Linux.

When the PoA is first switched on, the PoA MIH user sends a unicast registration request to the MIIS server and registers with the other PoAs in the network. The IP address and the MIHF ID of the MIIS are provided statically through a configuration file. Instead the other MIHF peers are dynamically discovered, by means of a UDP broadcast message with a broadcast MIHF ID in the destination MIHF field of the MIH_Register.request. Each MIH_USR_AP, which receives a MIH_Register.request from a remote peer, adds the new MIHF to the list of registered peers and completes the registration process by means of a unicast MIH_Register.response. Thus, the sender of the registration request is informed about the presence of other PoAs. After the registration is completed, the MIIS server sends a MIH_Push_Information.request to the MIH_USR_AP to inform it about supported RDF schemas.

Afterwards, the PoA MIH user creates a MIH_Get_Information.request containing a SPARQL query to obtain the MIHF ID and the link address (e.g., MAC address) pair of each PoA in the network. This information is useful when an associated MN completes a handover, because, as previously described, the MIH_USR_AP has to notify the previous serving PoA of that MN about handover completion (i.e., in case the MN could not notify it due to excessive link quality degradation or link broken).

The notification consists of a MIH_N2N_HO_Complete.request message, whose creation is triggered on receiving a MIH_MN_HO_Complete.request from the MN. However, the message sent by the MN contains only the link address of the old serving PoA but not its MIHF ID. For this reason, the PoA MIH user utilizes the information retrieved from the MIIS server, to associate a link address to a MIHF ID.

Let us see what happens when an MN completes association with a PoA, which now becomes the serving PoA. After receiving the MIH_Register.request from the MN, the MIH_USR_AP sends a MIH_Capability_Discover.request to the MN in order to discover supported MIH services and available interfaces. Afterwards, it updates its current load. When the PoA MIH user receives the MIH_Capability_Discover.response from the MN, it registers to the link paramenters events, i.e., MIH_Link_Going_Down and MIH_Link_Paramters_Report, if supported. Then, it sends a MIH_Link_Action.request to switch off all the unused wireless interfaces, if the MN has multiple interfaces, so as to reduce power consumption.

Optionally, the serving PoA can set a threshold on the signal level of the associated wireless interface, so as to be kept informed on changing conditions that might lead to a handover. In any case, at least one of the link parameter events (i.e., MIH_Link_Going_Down and MIH_Link_Paramters_Report) must be supported by the MIHF_MN, because, otherwise, the serving PoA can not know when handover is needed, and so can not assist the MN on handover.

The handover procedure is triggered the reception of either a MIH_Link_Going_Down or a MIH_Link_Paramters_Report indication message. This notification is sent by the MN when the link quality is degrading and, hence, a handover is imminent, or when the signal level of the associated interface crosses one of the thresholds set by the serving PoA. Afterwards, the MIH user of the serving PoA sends the MIIS server the MIH_Get_Information.request, which contains the list of SPARQL queries used to update the utilization table on the MIIS server and to obtain candidate PoAs for that MN. The ordered list of candidate PoAs is sent by the MIIS server by means of a MIH_Get_Information.response to the serving PoA, which parses the query result encoded in XML, obtains the ordered list of PoAs and creates the MIH_Net_HO_Commit.request.

Finally, this message, which contains the list of candidate PoAs, is sent to the MIH_USR_MN. The MN tries all the PoAs contained in MIH_Net_HO_Commit.request until association succeeds with one of them.

The old serving PoA will be informed about handover completion, by means of a MIH_Net_HO_Commit.response from the MN, if the link is still active, or by a MIH_N2N_HO_Complete.request from the new serving PoA. Afterwards, it updates its current load and completes the de-registration process with the MN towards a MIH_DeRegister.request and a MIH_Event_Unsubscribe.request. Note that these messages may not be received by the MN if the quality of the link is degraded or the link is broken.

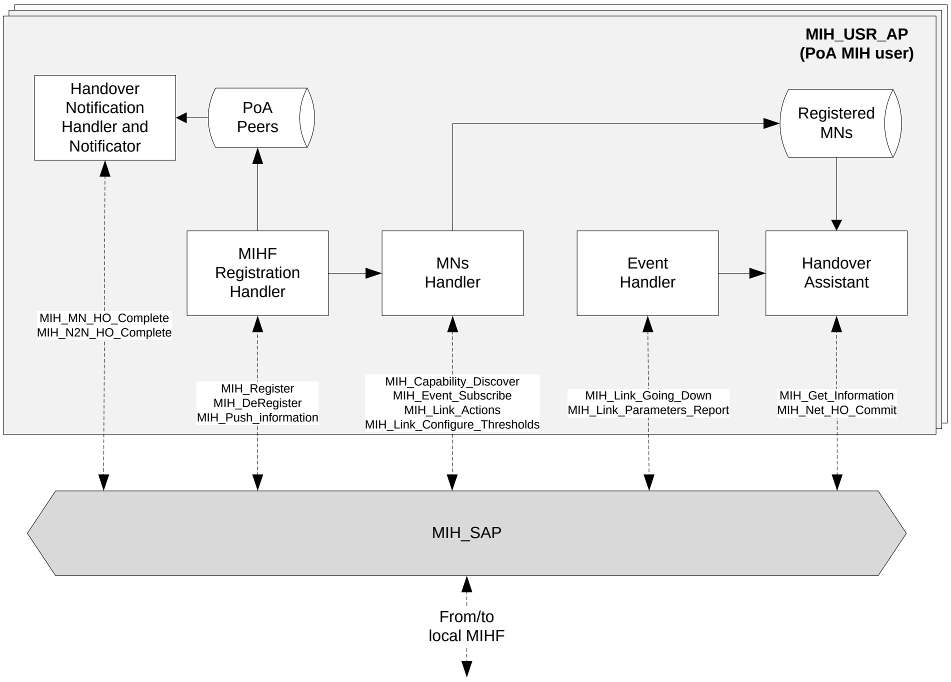

Figure 5.3: PoA MIH user functional scheme

As shown in Figure 5.3, the MIH_USR_POA is composed of the following logical modules:

- *MIHF Registration Handler*, which deals with registration and de-registration of new MIHF entities.

- *MNs Handler*, which is responsible for all the operations needed to register a new MN. After MIHF registration, it discovers the supported MIH services and the available interfaces on the MN. This information is stored in an internal data structure and is subsequently used by the *Handover Assistant* to assist the MN on handover. Afterwards, it registers the PoA MIH user to the link paramenters events and switches off the unused interfaces. Optionally, it can set a threshold on the associated wireless interface.

- *Event Handler*, which waits for event notification from the associated MNs. When it receives a new notification, it means that handover is imminent for that MN. This way, it contacts the *Handover Assistant*, which will query the MIIS server to obtain the candidate PoAs list.

- *Handover Assistant*, which assists associated MNs during handover. The handover process is triggered by the *Event Handler*, which communicates the MN that needs handover to the Handover Assistant. At this point, on the bases of the information contained in the internal data structure, it prepares the SPARQL/Update and SPARQL queries, and sends them in the MIH_Get_Information.request to the MIIS server . This way, it obtains the candidate PoAs list, which is finally passed to the MIH_USR_MN.

- *Handover Notification Handler and Notificator*, which is responsible for communicating the handover completion to the MIIS server and to the old serving PoA. Moreover, it manages notification from other PoAs and uses the paired MIHF ID and the link address information, stored in the internal data structure, to contact the other PoAs.

The data structure of the *Registered MNs* contains the list of network interfaces for each associated MN, so as to support it when handover is needed. Instead the data structure of the *PoA Peers* contains the paired list of link addresses and MIHF IDs for each registered PoA peer.

5.1.4 MN architecture

Each MN is composed of three modules, as shown in Figure 4.3, i.e., the MIH_LINK_SAP_MN, the MIHF_MN and the MIH_USR_MN. The main function of the MIH user is to start the bootstrap phase when the MN is first switched on, and, when handover is needed, to try all the candidate PoAs until an association succeeds.

Given the limited number of functions performed by the MIH user and in order to simplify the architecture of the MN, the MIH_USR_MN and the MIH_LINK_SAP have been combined. This way, the MIH_USR_MN no longer exists on the MN since its tasks are now performed by the MIH_LINK_SAP_MN, whose functional view is shown in Figure 5.4.

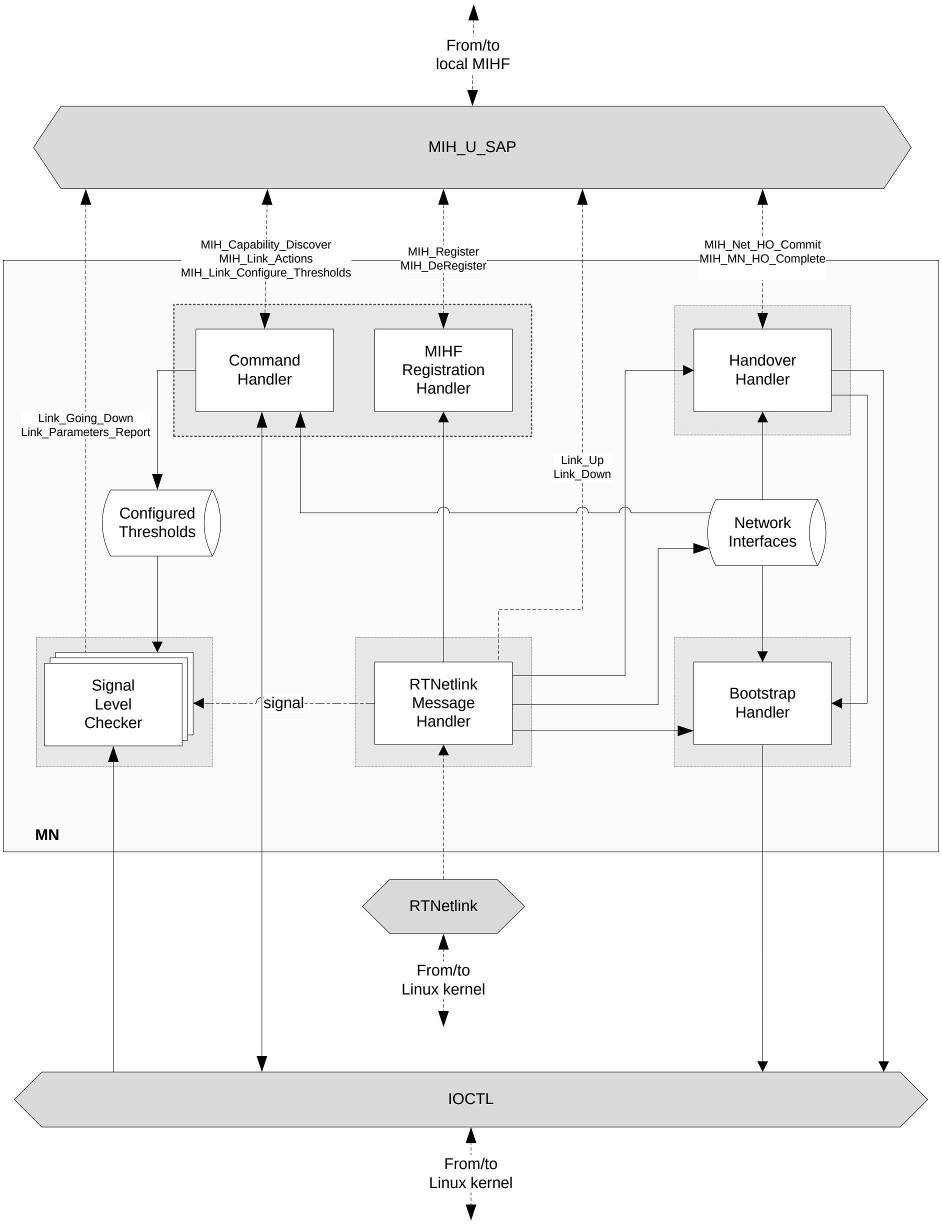

Figure 5.4: MIH_LINK_SAP_MN functional scheme

The MIH_LINK_SAP_MN is a link-layer dependent MIH_LINK_SAP for MNs with support for IEEE 802.11 wireless interfaces. It has been developed for Linux, using the RTNetlink user space/kernel interface, and it is composed of the following modules:

- *MIHF Registration Handler*, which deals with registration and de-registration of new MIHF entities. It is informed by the *RTNetlink Message Handler* when an association with a new PoA has been completed, so as to start the registration process with the serving PoA. Moreover, it manages the de-registartion process with the old serving PoA, when the MN completes a handover.

- *Command Handler*, which receives and processes MIH commands from the serving PoA. For instance, it is responsible for powering on/off the network interfaces and commanding a wireless scan, using ioctl system calls, as well as setting a threshold on the signal level of the associated interface. This threshold is then stored in the appropriate data structure, so as to be used by the *Signal Level Checker* module. Moreover, it responds to the MIH_Capability_Discover.request, using information stored in the internal data structure.

- *RTNetlink Message Handler*, which manages events triggering, analyzing messages from the Linux kernel through the RTNetlink interface. RTNetlink provides an effective way of gathering and controlling kernel media-independent network information. RTNetlink is a subset of Netlink which provides full-duplex communicatiosn between kernel modules and user-space processes through AF_NETLINK sockets. The information provided by RTNetlink can be classified into link-layer interface settings, network-layer interface settings, network-layer routing tables and rules, queuing settings, traffic classes and traffic filters. Via RTnetlink, this module can get all these elements of information. RTNetlink enables joining various multicast groups for getting notifications about changes in states and parameters from all the information classes mentioned above. This spares user-space applications from having constantly to poll the kernel for new information and allows for an elegant MIHF implementation. This way, RTNetlink Message Handler generates IEEE 802.21 events (i.e., Link_Up and Link_Down), every time there is a change in the state of a network interface. When one interface establishes L2 connectivity, this module is responsible for activating the thread, which checks the signal level on the specific interface. Moreover, it deals with update of the information contained in the internal data structure when there is a change in the state of an interface. If, for any reason, the associated interface loses connectivity, RTNetlink Message Handler triggers the beginning of a new bootstrap phase. Finally, it is responsible for informing the *Handover Handler* module of handover completion.

- *Signal Level Checker*, which controls the signal level of the IEEE 802.11 wireless interfaces. For each available interface, a specific thread is responsible for checking the RSSI level through the Linux Wireless Extension (WE) API. WE is a generic interface for obtaining the statistics and configurations of a WiFi link. WE methods are based on a set of ioctl calls and the */proc* file system. WE provides both dynamic and static information about the WiFi card and the associated link condition. A straightforward way to identify a network interface as a WiFi is to test if WE exists for that particular interface. The centralized entry */proc/net/wireless* provides wireless specific dynamic statistics from the driver that can be classified into two categories: quality and discarded packets. This entry is used indirectly via WE. Link quality statistics include the quality, level and noise of the link. Link level, indicating the current signal strength, and link noise level are measured at the receiver and are given either in *dBm* or in relative values. The one byte long signal strength value indicating the received Radio Frequency (RF) energy is defined as RSSI. WE is also used by the the MIH_LINK_SAP_MN, when the MN is first switched on, to obtain the available wireless interfaces, and this information is stored in the apposite data structure. If the signal level crosses a configured threshold, this module generates a Link_Parameters_Report event, which will be delivered to the local MIHF. Moreover, this module is responsible for generating the Link_Going_Down event when the quality of the link is degrading, and, hence, handover is needed. Note that if an interface is down or not associated, the thread responsible for checking its signal level, is not active. It will be woken up by the *RTNetlink Message Handler*, when the interface establishes L2 connectivity, so as to mimimize waste of resources.

- *Bootstrap Handler*, which manages the bootstrap phase. It is active, either when the MN is first switched on or if the currently associated interface loses connectivity.

- *Handover Handler*, which deals with the handover process. This is triggered as soon as a MIH_Net_HO_Commit.request is received from the serving PoA. Afterwards, this module tries all the candidate PoAs in the ordered list until an association succeeds with one of them. If, for any reason, the association failS with all the PoAs, the module contacts the *Bootstrap Handler* module, so as to start a new bootstrap phase. Notification of handover completion is received from the *RTNetlink Message Handler*. Finally, it creates the MIH_MN_HO_Complete.request message to notify the new serving PoA of handover completion, and tries to send the MIH_Net_HO_Commit.response to the old serving PoA.

Having combined the MIH_USR_MN and the MIH_LINK_SAP, there is a need for a modified SAP which unifies the services supplied by the MIH_SAP and the MIH_LINK_SAP. For this reason, the MIH_U_SAP was developed.

5.1.4.1 MIH_LINK_SAP extensions

The MIH_LINK_SAP_MN has been developed for Linux and it supports IEEE 802.11 wireless interfaces. It must be extended in order to support other technologies. However, due to the architecture of the MIH_LINK_SAP_MN, it requires only a few modification, so as to enable, for instance, WiMAX support. The MIIS server and the PoA MIH user, being technology-independent, do not require any modification.

5.1.5 ODTONE MIHF modifications

The implemented MIH users and MIH_LINK_SAP do not work with the MIHF provided by the latest release of ODTONE. Therefore in order to support the network-assisted handover procedures described above, some necessary modifications are needed. The following subsections give an overview of the modifications made to the MIHF provided by the ODTONE project.

5.1.5.1 MIHF registration

The function responsible for registering MIHF peers is not present in ODTONE. A new function was developed to enable an MIHF to send either a unicast or a broadcast MIH_Register.request message to remote peers, as well as to perform the registration. After registration completion, the two registered MIHF entities can communicate each others and use the services provided by the other peer.

5.1.5.2 Capability discover

The MIHF has to interact with the media-specific SAPs, by means of the MIH_LINK_SAP, in order to obtain the list of available interfaces on a device. In the current release of ODTONE, this information is statically taken from a configuration file. For this reason, modifications was made in order to support the MIH_Capability_Discover, so as to get available interfaces directly from the device.

5.1.5.3 Event subscription

In the latest release of ODTONE, an MIH user can subscribe multiple times to the same event on a specific interface. This way, when the subscribed event occurs, multiple notifications are sent to the subscriber. To avoid this situation the function responsible for event subscription was modified. The following MIH events are now supported:

- *MIH_Link_Up*, which is generated when the interface establishes L2 connectivity.
- *MIH_Link_Down*, which is generated when the interface is turned off.

- *MIH_Link_Going_Down*, which is generated when the signal level becomes lower than a configured threshold.

- *MIH_Link_Parameters_Report*, which is generated when the signal level of a specific interface crosses a threshold configured by a MIH_USR_AP towards a MIH_Link_Configure_Thresholds.request.

5.1.5.4 Transaction handler

An MIH transaction is identified by a sequence of messages with the same Transaction ID exchanged between two remote MIHF entities, as described in 2.6.1. This identifier is created at the node initiating the transaction and it is carried over in the header of the MIH protocol frame to match a request message with its corresponding response message.

A new transaction handler was developed, so as to be fully compliant to the IEEE 802.21 standard. Each response message has the same Transcation ID of the corresponding request and this identifier is never modified during message exchanges. Moreover, it was added the support for broadcast request messages.

5.1.5.5 Command support

Finally, it has been added complete support for the following messages:

- MIH_Capability_Discover.

- MIH_Register and MIH_DeRegister.

- MIH_Get_Information and MIH_Push_Information (with RDF support).

- MIH_Link_Configure_Thresholds.

- MIH_Link_Actions (which supports power down, disconnect, wireless scan and link up actions).

- MIH_N2N_HO_Complete and MIH_MN_HO_Complete.

- MIH_Net_HO_Commit.

5.2 Testbed scenario and assumptions

In order to validate the proposed solution, a testbed has been set up, consisting of the following entities:

- 3 PoAs, with co-located PoSs, two of which equipped with an IEEE 802.11a wireless network interface and one with an IEEE 802.11g wireless network interface.
- An MIIS server, supporting RDF representation and SPARQL queries.
- A mobile node, supporting IEEE 802.11 technology.

Figure 5.5: ALIX embedded PCs used in the testbed scenario

Figure 5.6 shows the testbed scenario. All PoAs consist of ALIX embedded PCs[8] equipped with Voyage Linux OS distribution[9]. The MIIS server was installed on a dedicated PC with standard hardware, connected to the PoAs via a full mesh Ethernet LAN. This is to allow network-to-network exchanges to favor the handover and inform the interested entities of handover completion, and it is done by means of discovery procedures.

With regard to the MN, experiments were run both with a laptop equipped with a single IEEE 802.11a/g wireless interface and with an ALIX embedded PC equipped with two interfaces (i.e., IEEE 802.11a and IEEE 802.11g). PoAs have been put in the topology shown in Figure 5.6, with a few meters overlapping coverage areas. It assumed that the MN and the serving PoA communicates toward the wireless channel, while the PoA and the MIIS server use the wired connection. Messages exchanged during the tests are those shown in Figure 4.4.

Basic connectivity during handover has been successfully validated in the testbed scenario shown in Figure 5.6, which also contains the path followed by the MN during the test. However, these results are qualitative and hence are not reported here.

When the MN is first switched on, it scans through all the interfaces to find a suitable PoA. After association and IP establishment, the MN starts to move.

---

[8] http://www.pcengines.ch/.

[9] http://linux.voyage.hk/.

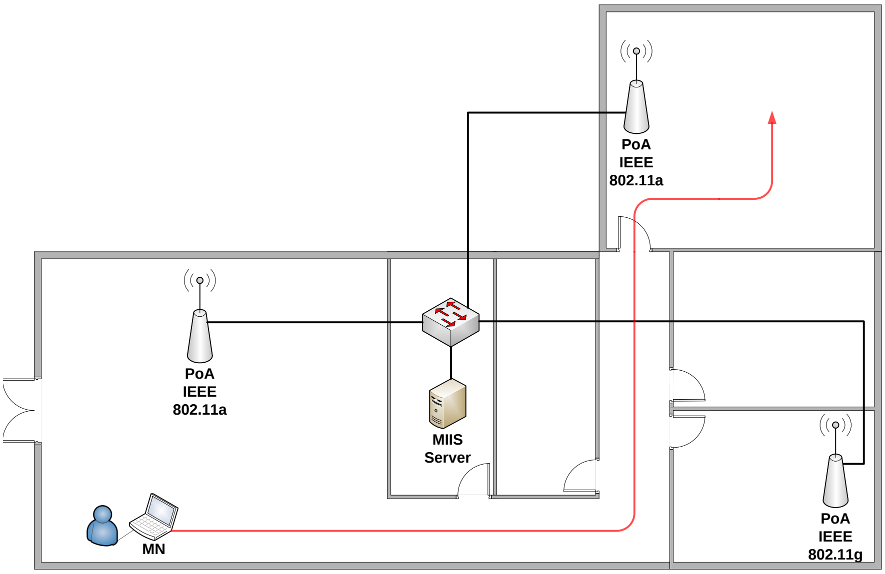

Figure 5.6: Testbed scenario

Afterwards, it automatically switch among available PoAs in a seamless way and without any user's intervention.

The handover execution (e.g., L2 and higher layers signaling) is out of the scope of the IEEE 802.21 standard and needs to be addressed by any mobility management protocols, for instance, so as to guarantee session continuity. In order to overcome this problem, in this testbed scenario each PoA automatically assign a pre-configured IP address to the MN when it completes the handover.

5.3 Results

Noticeable differences have not been found for the two types of MN (i.e., the laptop and the ALIX embedded PC), hence only the results obtained with the laptop are reported. In the following sections, results of two experimental tests are shown. First has been used to show that the proposed solution is promising to keep the handover latency small. On the other hand, the latter has been used to show the maximum number of MNs that the MIIS server is able to manage.

5.3.1 Handover latency

In order to obtain more quantifiable and repeatable results, PoAs were put in proximity of one another in a laboratory and the handover of the MN was triggered via a software indication (i.e., a MIH_Link_Going_Down.indication message) through an automated test procedure.

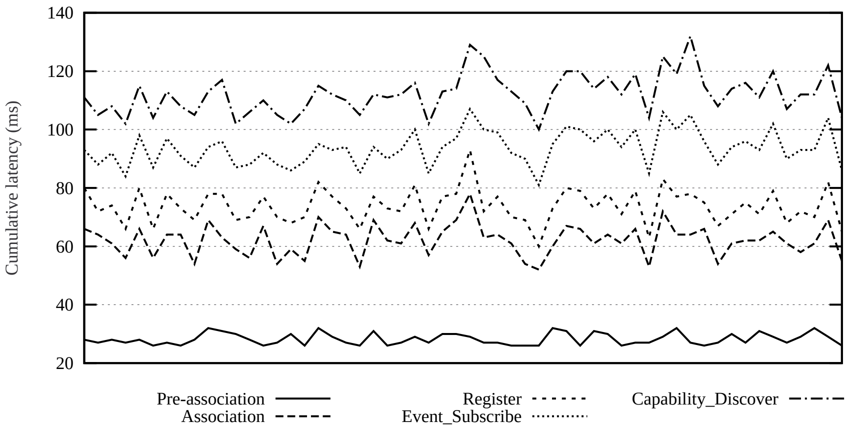

Figure 5.7: Latency of the handover phases in different experiments

In Figure 5.7 is reported the latency of the different phases of the handover, in a cumulative manner, i.e., the latency of each phase is added to that of the previous phase. The samples obtained from more than 50 tests are included in the plot. The phases included in Figure 5.7 are the following:

- *Pre-association phase*, which starts when the MN sends a MIH_Link_Going_Down.indication to the serving PoA, for indicating that the link quality is degrading and, hence, a handover is imminent. It also includes message exchanges to and from the MIIS server (i.e., MIH_Get_Information messages) for obtaining information about neighboring PoAs. This phase ends when the MN receives the MIH_Net_HO_Commit.request message from the serving PoA with the list of candidate PoAs.
- *Association phase*, which is totally dependent on the link-layer functions. This phase indicates the time necessary to the MN for establishing L2 connectivity.
- *Register phase*, where the MIHF of the MN discovers the MIHF of the PoA and binds it to itself towards MIH_Register messages. This way, the MN declares its presence to the associated PoA, so as to be later supported on handover decisions.
- *Capability Discover phase*, which represents message exchanges between the MN and the serving PoA (i.e., MIH_Capability_Discover messages). This message is used by the serving PoA to discover the supported MIH services and the available interfaces on the MN.
- *Event Subscribe phase*, where the serving PoS registers to the link parameters events, by means of a MIH_Link_Event_Subscribe.request, so as to be kept informed on changing condition that might lead to a handover.

As can be seen, the greatest contribution comes from the association phase, which is totally dependent on the link-layer functions. For instance, in the specific case of IEEE 802.11, it has been demonstrated[10] that the association time can be reduced significantly with some minor modifications to the devices. On the other hand, the pre-association phase is very small, mostly because there are no other concurrent handovers.

Note that in the handover latency are not considered node-to-network and network-to-network messages used to notify the handover completion, as well as the command to switch off all the unused wireless interfaces. In fact, these messages do not contribute to the handover latency.

5.3.2 MIIS server stress test

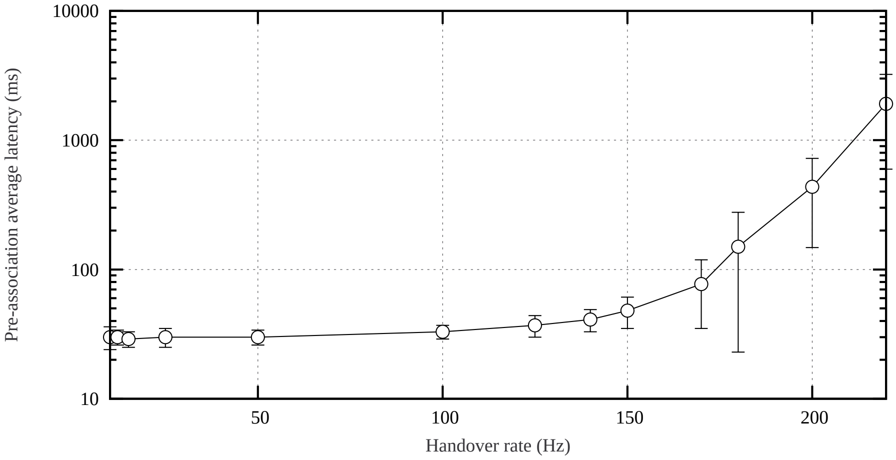

Figure 5.8: Average latency of the pre-association phase vs. MIIS load

Therefore, it has been decided to artificially inflate the MIIS server load, by injecting 50 simulated PoA in the network, querying the MIIS server at random times. Each simulated PoA sent to the MIIS server a MIH_Get_Information.request message, containing a SPARQL/Update query used to keep its current load up-to-date, as well as a SPARQL query used to obtain the list of candidate PoAs.

In Figure 5.8 is reported the pre-association latency when the total number of handover procedures per second, on average, is increased from 10 to 220. In fact, this phase includes querying the MIIS server and so it is the only affected by this test.

As can be seen, the latency remains almost constant until 150 handovers per second, i.e., until the implemented MIIS server is able to dispatch the MIH_Get_Information.response messages with no significant delay. If the load increases even further, the latency increases very sharply because the MIIS server eventually becomes saturated. It is worth noting that, with this implementation, a standard PC is able to manage a rate of up to 150 handovers per second, i.e., hundreds to thousands of MNs, depending on their degree of mobility and on the coverage of the technologies employed.

In order to improve the performances of the MIIS server, it is possible to reduce its service area so as to distribute its load on several network entities.

5.4 Conclusions

A proof-of-concept experimental evaluation, with overlapping non-interoperable IEEE 802.11a and IEEE 802.11g networks, has been carried out to show the feasibility and effectiveness of the proposed solution, as well as the scalability of the MIIS server. Preliminary experimental results show that the proposed solution is promising to keep handover latency down while removing the burden of scanning on the mobile device. Moreover, the designed and implemented MIIS server is able to manage several handovers per second, i.e., hundreds to thousands of MNs, depending on their degree of mobility and on the coverage of the technologies employed.

Chapter 6

Conclusions

In this thesis a solution has been designed and implemented to enable network-assisted handover using the IEEE 802.21 standard so as to reduce the handover latency and the energy consumption of mobile devices. In fact, an MN needs to do a wireless scan only the first time it enters a network. Afterwards the mobile device is instructed by the network itself on the best possible serving access points, thus removing the need for scanning. Benefits are especially significant if multiple radios coexist in the same devices, since only one of them needs to be powered on at any time.

Moreover, the MIIS server, the PoA MIH user, the MN MIH user and the MIH_LINK_SAP of the MN have been designed, implemented and tested under Linux, using a modified version of the MIHF provided by the ODTONE project. Algorithms have also been conceived for selecting the best candidate PoAs for a MN involved in handover at a particular time and place. In fact, so as to promote competition by differentiation of equipment capabilities and services, the IEEE 802.21 standard leaves certain elements unspecified, e.g., the procedures and alghoritms to enable network-assisted handover, as well as the architectures and the functional structures of the MIH network entities (e.g., the MIIS server).

Altough the developed MIH_LINK_SAP supports only the IEEE 802.11 technology, few mofications need be done so as to extend the support for other technlogies (e.g., WiMAX). In fact, the MIIS server and the PoA MIH user are technology-independent, so they do not need to be modified to support other technologies.

Vendor-specific IEs and containers of IEs have been defined in order to enable the network-assisted handover solution described above, as well as to construct neighboring relationships among the PoAs in the network.

A prototype implementation has been developed to prove the feasibility of the solution. Preliminary experimental results show that the proposed solution is promising to keep handover latency down while removing the burden of scanning on the MN. Moreover, the designed and implemented MIIS server is able to manage several handovers per second, i.e., hundreds to thousands of MNs, depending on their degree of mobility and on the coverage of the technologies employed.

6.1 Outlook

In this current implementation, a baseline simple algorithm is employed, which seeks the candidate PoAs with their utilization above a configurable threshold and sorts the remaining ones in decreasing order of their weights in the connectivity graph. In order effectively to find the next best PoA for a MN, a more evoluted algorithm may be implemented, which is able to predict the movement of the MNs, for instance, using information from a GPS interface.

Moreover, an important challenge facing the IEEE 802.21 is the unification of all media-specific technologies under one abstract interface (i.e., MIH_LINK_SAP). This approach may be difficult to realize in practice within a short period of time due to the large number of technology-specific standards within and outside the IEEE 802 systems, which must be extended to conform to the MIH_LINK_SAP. In certain technologies, media-specific primitives may be already available and is only required the correctly mapping of them to MIH_LINK_SAP primitives.

Other technologies, however, may require extensions to media-specific primitives. The IEEE 802.21 WG has established liaisons with the IEEE 802.11 TG "u" and the IEEE 802.16 TG "g". To support MIH services, appropriate primitives will be added to the IEEE 802.11 and the IEEE 802.16 standards. Proposals have been made to the 3GPP Standards Association to incorporate MIH services so as to support handover between WiMAX, 3GPP and LTE networks. Although the inclusion of these proposals into the Release-8 specification for the 3GPP was not accepted, MIH remains an attractive technology for enhancing IP mobility across heterogeneous accesses for future 3G releases. Furthermore, the IETF MIPSHOP WG is specifying an higher layer transport for the MIH protocol and mechanisms to discover MIHF peers.

Another challenge to the widespread adoption of the IEEE 802.21 is the lack of a conformance statement detailing the mandatory set of primitives and primitive sequences required to realize a particular use case. Such a statement would provide a method to verify that IEEE 802.21-based equipments conform to the standard and assurances to the community that equipments from different vendors will interoperate with each other.

The IEEE 802.21 standard must also address a number of additional features to ensure acceptance. Some features are better left to individual companies to be addressed, giving them an opportunity to distinguish themselves in the marketplace. However, if a required set of common hooks and interfaces is not in place and interoperability is not ensured, industry adoption will be difficult.

Important features to the deployment of MIH services, such as MIIS provisioning, MIH security and multi-radio power management are not fully addressed in the specification yet. MIIS provisioning deals with issues concerning how information is populated to and stored in the MIIS server. MIH security includes mechanisms to protect MIH protocol messages based on mutually authenticating MIH entities. Because operating multiple radios can be a significant drain on the battery of a device, mechanisms must be in place to facilitate better power management for multi-radio devices.

In summary, the success of this standard depends not only upon the activities within the IEEE 802.21 WG, but also upon the acceptance of this technology by other standards and industry forums. To achieve acceptance and wide deployment in the future, additional specifications describing use case scenarios with requirements, features and required extensions to media-specific technologies are required. This means continued effort on the part of the IEEE 802.21 WG, as well as close collaboration and liaison with other standard organizations.

6.2 Acknowledgment

The research leading to these results has been partially funded by the European Communitys Seventh Framework Programme (FP7/2007-2013) under Grant Agreement n.233679. The SANDRA project is a Large Scale Integrating Project for the topic AAT.2008.4.4.2 (Integrated approach to network centric aircraft communications for global aircraft operations). The project has 31 partners and started on 01/10/09.

This work has been carried out during my internship at Intecs S.p.a., in the division Automotive and Telecommunications. I would like to thank Raffaella Mambrini as the business unit leader for her support, and all my team co-workers for their help and the insightful discussions on the topic.

Appendices

Appendix A

RDF Extended Schema

This RDF extended schema contains the definition of the vendor-specific IEs and containers of IEs described in 4.1.2, which were defined to support the network-assisted handover solution described in Chapter 4. In the following the tags in bold must be substituted with the actual values, depending on the system parameters.

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
  <!ENTITY mihbasic "BASIC_SCHEMA_URL_TO_BE_ASSIGNED">
  <!ENTITY mihextended "EXTENDED_SCHEMA_URL_TO_BE_ASSIGNED">
  <!ENTITY owl "http://www.w3.org/2002/07/owl#">
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
]>

<rdf:RDF xmlns:rdf="&rdf;"
  xmlns:rdfs="&rdfs;"
  xmlns:mihbasic="&mihbasic;"
  xml:base="&mihbasic;"
  xmlns:mihextended="&mihextended;"
  xmlns:owl="&owl;"
  xmlns:xsd="&xsd;"
>

<owl:Ontology rdf:about="">
  <rdfs:label>
      Extended Schema for IEEE 802.21 Information Service
  </rdfs:label>
  <owl:versionInfo>1.0</owl:versionInfo>
</owl:Ontology>

<owl:ObjectProperty rdf:ID="ie_poa_load">
  <mihbasic:ie_identifier>
```

```
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="&mihbasic;POA"/>
  <rdfs:range rdf:resource="&xsd;double"/>
  <rdfs:comment>
      Represents the current load on the PoA.
    It is between 0 and 1
  </rdfs:comment>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie_poa_mihf_id">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="&mihbasic;POA"/>
  <rdfs:range rdf:resource="&xsd;string"/>
  <rdfs:comment>
    Represents the MIHF ID of the PoA
  </rdfs:comment>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie_poa_querier_id">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="&mihbasic;POA"/>
  <rdfs:range rdf:resource="&xsd;string"/>
  <rdfs:comment>
    MIHF ID of the MN involved in handover
  </rdfs:comment>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie_container_list_of_neighbor_poas">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="&mihbasic;POA"/>
  <rdfs:range rdf:resource="#LIST_OF_NEIGHBOR_POAS"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="LIST_OF_NEIGHBOR_POAS">
  <rdfs:subClassOf>
    <owl:Restriction>
    <owl:onProperty rdf:resource="#ie_container_neighbor_poa"/>
    <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">0
    </owl:minCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

```
<owl:ObjectProperty rdf:ID="ie_container_neighbor_poa">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="#LIST_OF_NEIGHBOR_POAS"/>
  <rdfs:range rdf:resource="#NEIGHBOR_POA"/>
  <rdfs:comment>
      This class contains information about a neighboring PoA
  </rdfs:comment>
</owl:ObjectProperty>

<owl:Class rdf:ID="NEIGHBOR_POA">
  <rdfs:subClassOf>
    <owl:Restriction>
    <owl:onProperty rdf:resource="#ie_link_addr"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
    <owl:onProperty rdf:resource="#ie_weight"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1
    </owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="ie_link_addr">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="#NEIGHBOR_POA"/>
  <rdfs:range rdf:resource="&mihbasic;LINK_ADDR"/>
  <rdfs:comment>
    The neighboring PoA link address
  </rdfs:comment>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie_weight">
  <mihbasic:ie_identifier>
    ID_TO_BE_ASSIGNED
  </mihbasic:ie_identifier>
  <rdfs:domain rdf:resource="#NEIGHBOR_POA"/>
  <rdfs:range rdf:resource="&xsd;unsignedInt"/>
  <rdfs:comment>
    The weight value used for constructing the connectivity graph.
  </rdfs:comment>
```

```
</owl:DatatypeProperty>

</rdf:RDF>
```

The IEEE 802.21 standard has reserved a set of IE namespace ranges for development and testing. These have been used in experimental evaluations but should not be used in released products. In fact, vendors shall specify their own IEs using the namespace allocated to them. This way , the following tags must be defined:

- Basic and extended schema URLs.
- ie_poa_load identifier.
- ie_poa_mihf_id identifier.
- ie_poa_querier_id identifier.
- ie_container_list_of_neighbor_poas identifier.
- ie_container_neighbor_poa identifier.
- ie_link_addr identifier.
- ie_weight identifier.

Acronyms

3G	Third Generation
3GPP	Third Generation Partnership Project
3GPP2	Third Generation Partnership Project 2
4G	Fourth Generation
ACK	Acknowledgement
AID	Action Identifier
AP	Access Point
API	Application Programming Interface
BS	Base Station
C-SAP	Control-SAP
COTS	Commercial, Off-The-Shelf
DHCP	Dynamic Host Configuration Protocol
DO	Downlink-Only
DSL	Digital Subscriber Line
DVB	Digital Video Broadcasting
EAP	Extensible Authentication Protocol
FQDN	Fully Qualified Domain Name
GAN	Generic Access Network
GPL	General Public License
GPRS	General Packet Radio Service
GPS	Global Positioning System
HESSID	Homogeneous Extended Service Set ID
ID	Identifier
IE	Information Element
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
IPTV	Internet Protocol Television
LAN	Local Area Network
LGPL	Lesser General Public License
LLC	Logical Link Control
LTE	Long Term Evolution

L2	Layer 2
L3	Layer 3
L7	Layer 7
MAC	Medium Access Control
MediaFLO	Media Forward Link Only
MICS	Media Independent Command Service
MIES	Media Independent Event Service
MIH	Media Independent Handover
MIHF	Media Independent Handover Function
MIIS	Media Independent Information Service
MIME	Multipurpose Internet Mail Extensions
MIP	Mobile IP
MIPL	Mobile IPv6 for Linux
MIPSHOP	Mobility for IP Performance, Signaling, and Handoff Optimization
MN	Mobile Node
M-SAP	Management-SAP
MSGCF-SAP	MAC State Generic Convergence Function - SAP
NAI	Network Access Identifier
OSI	Open Systems Interconnection
PC	Personal Computer
PDA	Personal Digital Assistant
PDU	Protocol Data Unit
PHY	Physical
PoA	Point Of Attachment
PoS	Point Of Service
PPP	Point-to-Point Protocol
PSTN	Public Switched Telephone Network
QoE	Quality Of Experience
QoS	Quality Of Service
RDF	Resource Description Framework
RF	Radio Frequency
RSSI	Received Signal Strength Indication
SAP	Service Access Point
SCTP	Stream Control Transmission Protocol
SID	Service Identifier
SIP	Session Initiation Protocol
SINR	Signal to Interference plus Noise Ratio
SSID	Service Set Identification
TCP	Transmission Control Protocol
TG	Task Group
TLV	Type-Length-Value
T-DMB	Terrestrial Digital Multimedia Broadcasting
UDP	User Datagram Protocol
UIR	Unauthenticated Information Request

UMA	Unlicensed Mobile Access
UMTS	Universal Mobile Telecommunications System
URL	Uniform Resource Locator
UTRAN	UMTS Terrestrial Radio Access Network
VC	Vehicular Communication
VOD	Video-on-Demand
VoIP	Voice over IP
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
WAN	Wide Area Network
WE	Wireless Extension
WG	Working Group
WiFi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
XML	Extensible Markup Language

Bibliography

[1] K. Pentikousis, R. Aguero, J. Gebert, J. A. Galache, O. Blume, and P. Paakkonen, “The Ambient Networks Heterogeneous Access Selection Architecture,” in *First Ambient Networks Workshop on Mobility, Multiaccess, and Network Management (M2NM)*, October 2007, pp. 49–54.

[2] J. Makela, K. Pentikousis, and V. Kyllonen, “Mobility Trigger Management: Implementation and Evaluation,” *International Journal of Communications, Network and System Sciences*, vol. 2, no. 3, pp. 211–221, June 2009.

[3] IEEE, “IEEE Standard for Local and metropolitan area networks– Part 21: Media Independent Handover Services,” *IEEE Std 802.21–2008*, January 2009.

[4] L. Eastwood, S. Migaldi, X. Qiaobing, and V. Gupta, “Mobility Using IEEE 802.21 in a Heterogenous IEEE 802.16/802.11-Based, IMT-Advanced (4G) Network,” *IEEE Wireless Communications*, vol. 15, no. 2, pp. 26–34, April 2008.

[5] S. Das, M. Tauil, Y.-h. Cheng, A. Dutta, D. Baker, M. Yajnik, and D. Famolari, “IEEE 802.21: Media Independent Handover: Features, Applicability, and Realization,” *IEEE Communications Magazine*, vol. 47, no. 1, pp. 112–120, January 2009.

[6] E. Piri and K. Pentikousis, “IEEE 802.21,” *The Internet Protocol Journal*, vol. 12, no. 2, pp. 7–27, June 2009.

[7] A. de La Oliva, A. Banchs, I. Soto, T. Melia, and A. Vidal, “IEEE 802.21 (Media Independent Handover services) Overview,” *IEEE Wireless Communications*, vol. 15, no. 4, pp. 96–103, August 2008.

[8] G. Lampropoulos, A. K. Salkintzis, and N. Passas, “Media-Independent Handover for Seamless Service Provision in Heterogeneous Networks,” *IEEE Communications Magazine*, vol. 46, no. 1, pp. 64–71, January 2008.

[9] S. Giordano, D. Lenzarini, A. Puiatti, M. Kulig, H. Nguyen, and S. Vanini, “Demonstrating seamless handover of multi-hop networks,” in *REALMAN ’06: Proceedings of the 2nd international workshop on Multi-hop ad hoc*

networks: from theory to reality. New York, NY, USA: ACM, 2006, pp. 128–130.

[10] X. Chen and D. Qiao, "HaND : Fast Handoff with Null Dwell Time for IEEE 802.11 Networks," *IEEE INFOCOM 2010. 29th IEEE International Conference on Computer Communications*, pp. 1–9, March 2010.

[11] Y. Choi and S. Choi, "Service Charge and Energy-Aware Vertical Handoff in Integrated IEEE 802.16e/802.11 Networks," *IEEE INFOCOM 2007. 26th IEEE International Conference on Computer Communications*, pp. 589–597, May 2007.

[12] A. de La Oliva, T. Melia, A. Vidal, C. J. Bernardos, I. Soto, and A. Banchs, "A case study: IEEE 802.21 enabled mobile terminals for optimized WLAN/3G handovers," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 2, pp. 29–40, April 2007.

[13] Q. B. Mussabbir, W. Yao, Z. Niu, and X. Fu, "Optimized FMIPv6 Using IEEE 802.21 MIH Services in Vehicular Networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3397–3407, November 2007.

[14] G. Gehlen, E. Weiss, S. Lukas, C.-H. Rokitansky, and B. Walke, "Architecture of a Vehicle Communication Gateway for Media Independent Handover," *3rd International Workshop on Intelligent Transportation, WIT*, March 2005.

[15] F. Cacace and L. Vollero, "Managing Mobility and Adaptation in Upcoming 802.21 enabled devices," in *International Workshop on Wireless mobile applications and services on WLAN hotspots*, 2006, pp. 1–10.

[16] W.-S. Lim, D.-W. Kim, Y.-J. Suh, and J.-J. Won, "Implementation and performance study of IEEE 802.21 in integrated IEEE 802.11/802.16e networks," *Computer Communications*, vol. 32, no. 1, pp. 134–143, January 2009.

[17] E. Piri and K. Pentikousis, "Towards a GNU/Linux IEEE 802.21 Implementation," *IEEE International Conference on Communications*, pp. 1–5, January 2009.

[18] Y. Lopez and E. Robert, "OpenMIH, an Open-Source Media-Independent Handover Implementation and Its Application to Proactive pre-Authentication," *Mobile Networks and Management: First International Conference, MONAMI 2009*, October 2009.

[19] A. Dutta, S. Das, D. Famolari, Y. Ohba, K. Taniuchi, V. Fajardo, and T. Kodama, "Secured Seamless Convergence across Heterogeneous Access Networks," *World Telecommunication Congress*, 2006.

[20] T.-Y. Chung, Y.-M. Chen, P.-C. Mao, C.-K. Tsai, S.-W. Lai, and C.-P. Chen, "The Design and Implementation of IEEE 802.21 and Its Application on

Wireless VoIP," *IEEE Vehicular Technology Conference (VTC)*, pp. 1–5, May 2010.

[21] L. Sarakis, G. Kormentzas, and F. M. Guirao, "Seamless Service Provision for Multi Heterogeneous Access," *IEEE Wireless Communications*, vol. 16, no. 5, pp. 32–40, October 2009.

[22] IEEE, "IEEE Standard for Architectural Building Blocks Enabling Network-Device Distributed Decision Making for Optimized Radio Resource Usage in Heterogeneous Wireless Access Networks," *IEEE Std 1900.4–2009*, February 2009.

[23] V. Ghini, P. Salomoni, and G. Pau, "Always-Best-Served Music Distribution for Nomadic Users over Heterogeneous Networks," *IEEE Communications Magazine*, vol. 43, no. 5, pp. 69–74, May 2005.

[24] W.-S. Lim, D.-W. Kim, Y.-J. Suh, and J.-J. Won, "Efficient WLAN Discovery Schemes Based on IEEE 802.21 MIH Services in Heterogeneous Wireless Networks," *IEEE INFOCOM 2008. 27th IEEE International Conference on Computer Communications*, pp. 1–5, December 2008.